SUPER JEUX AMSTRAD

50 programmes de jeux en Basic

Jean-François Sehan



Autres ouvrages relatifs à Amstrad

— 102 programmes pour Amstrad - Jacques Deconchat
— Amstrad en famille - Jean-François Sehan
— Basic Amstrad : méthodes pratiques - Jacques Boisgontier

A paraître

— Clefs pour Amstrad - Daniel Martin
— La découverte de l'Amstrad - Daniel-Jean David
— Exercices en Basic pour Amstrad - Maurice Charbit
— Basic Plus - 80 routines sur Amstrad - Michel Martin
— Assembleur de l'Amstrad - Marcel Henrot

Ouvrages d'initiation

— Visa pour l'informatique - Jean-Michel Jégo
— Visa pour le Basic - Jean-Michel Jégo
— Mon ordinateur - Jean-Claude Barbance
— Le logotron informatique - Jean-Pierre Petit
— L'ordinateur apprivoisé - François Picard et Danielle Shaw
— Les mots de la micro - Alan Freedman, adapté par Bernard Sauteur

ISBN: 2-86595-257-6

SUPER JEUX AMSTRAD

50 programmes de jeux en Basic

Jean-François Sehan



Editions du PSI
1985

AVERTISSEMENT

Si vous voulez utiliser correctement les programmes de ce livre, lisez attentivement les conseils suivants :

■ Avant de taper un programme, vous devez initialiser Amstrad en appuyant simultanément sur les touches [SHIFT], [CTRL] et [ESC].
Effectuez la même opération avant de charger un programme à partir du datacorder.

■ Pour gagner en rapidité d'exécution certaines instructions NEXT ne sont pas suivies d'une variable (gain de 10 à 20 %).

■ Si vous n'obtenez pas les mêmes résultats que sur les exemples, vérifiez bien toutes les lignes et surtout celles contenant des DATA.

■ Dès que les programmes sont entrés dans votre AMSTRAD, sauvez-les sur le datacorder. Pour les programmes très longs, il est vivement conseillé d'effectuer une première sauvegarde au milieu de leur frappe. N'attendez pas qu'ils soient entièrement entrés et testés... une coupure de courant est si vite arrivée !

■ Quand vous entrez des chiffres décimaux, assurez-vous que vous tapez un point et non une virgule entre deux chiffres.

■ Assurez-vous que vous êtes en majuscules dès la mise en route, en appuyant sur CAPS LOCK.

SOMMAIRE

Chapitre III : Jeux de hasard

CHAPITRE I :
JEUX D'ADRESSE



GRENOUILLE

Si vous croyez que la vie d'une grenouille est de tout repos, vous allez vite changer d'idée.

Car l'héroïne de ce programme s'est endormie sur son nénuphar et rêve qu'elle se trouve au bord d'une route et d'une rivière qu'elle doit traverser pour regagner les rochers salutaires de l'autre rive.

Après avoir lancé le programme, vous pouvez manœuvrer votre grenouille latéralement avec les touches P et Q ; vous devez la faire sauter en utilisant la barre d'espacement. **Attention à bien viser entre les véhicules qui défilent impitoyablement** ; sinon vous perdrez à coup sûr une des trois vies qui vous sont allouées à chaque début de partie.

Ouf ! vous êtes sur le bord de la rivière. Il vous faut maintenant emprunter un tronc d'arbre ou un crocodile pour aller sauter sur l'un des rochers. Mais alors la pauvre grenouille se retrouve à nouveau sur le bord de la route. Son rêve ne cessera que lorsqu'elle aura regagné successivement les cinq rochers.

Alors, à vos bonds, prêt, partez !...

A la fin de chaque partie, relevez votre score (en haut à gauche de l'écran) et enfoncez la touche ENTER pour un autre jeu.

```
10 REM GRENOUILLE
20 REM ------------------------------
30 DEFINT A-Z:SYMBOL AFTER 128
40 DIM P(22),I(22):GOSUB 500
50 V=2:LOCATE 24,1:PAPER 0:PEN 1:PRINT "VIE: 2"
60 P=0
70 T=0
80 X=19:Y=22
90 PEN I(Y):PAPER P(Y):LOCATE X,Y:PRINT CHR$(138):
AX=X:AY=Y
100 A=2:FOR I=1 TO 8
110 PEN I(I*2+A):PAPER P(I*2+A):LOCATE 1,I*2+A:PRI
NT MID$(L$(I),D(I),40)
120 IF I=4 THEN A=4
130 NEXT
140 IF Y=2 THEN IF C<1 THEN 1100 ELSE 1200
150 FOR I=1 TO 7 STEP 2
160 D(I)=D(I)+1:IF D(I)=25 THEN D(I)=1
170 D(I+1)=D(I+1)-1:IF D(I+1)=0 THEN D(I+1)=24
180 NEXT
190 PEN 1:PAPER 0:LOCATE 5,1:PRINT "POINTS:";P
280 REM ------------------------------
290 REM DEPLACEMENTS
300 A$=INKEY$
310 IF A$="P" THEN X=X+1
320 IF A$="Q" THEN X=X-1
330 IF A$=" " THEN Y=Y-2:P=P+1
340 IF AY=12 OR AY=22 THEN PEN I(AY):PAPER P(AY):L
OCATE AX,AY:PRINT " "
350 C=TEST(X*16-2,404-Y*16)+TEST(X*16-14,404-Y*16)
360 IF Y<12 AND C>5 THEN 1100
370 IF Y>12 AND C>0 THEN 1100
380 IF X<2 OR X>39 THEN 1100
390 IF AY<>Y THEN 90
400 IF Y=4 OR Y=8 THEN X=X-1
410 IF Y=6 OR Y=10 THEN X=X+1
420 GOTO 90
480 REM ------------------------------
490 REM GRAPHIQUES
500 SYMBOL 128,63,72,136,255,128,255,136,112
510 SYMBOL 129,224,16,8,255,1,255,34,28
520 SYMBOL 130,7,8,16,255,128,255,68,56
530 SYMBOL 131,248,36,34,255,1,255,17,14
540 SYMBOL 132,254,142,142,254,255,255,68,56
550 SYMBOL 133,127,113,113,127,255,255,34,28
560 SYMBOL 134,15,23,35,65,65,35,23,15
570 SYMBOL 135,240,248,252,254,254,252,248,240
580 SYMBOL 136,240,232,196,130,130,196,232,240
590 SYMBOL 137,15,31,63,127,127,63,31,15
600 SYMBOL 138,28,73,127,28,28,62,99,99
610 SYMBOL 139,255,255,255,255,255,255,255,255
620 SYMBOL 140,255,255,255,255,255,255,68,56
630 SYMBOL 141,255,255,255,255,255,255,0,0
640 SYMBOL 142,0,192,96,48,31,13,7,255
```

```
650 SYMBOL 143,0,0,0,0,255,255,255,255
660 SYMBOL 144,0,0,0,0,192,240,252,255
670 FOR J=1 TO 4:A$="":READ N,M,O
680 FOR I=1 TO O
690 READ A:IF A=17 THEN A$=A$+" " ELSE A$=A$+CHR$(
A+128)
700 NEXT I
710 FOR I=1 TO M:L$(N)=L$(N)+A$:NEXT I
720 NEXT J
730 L$(3)=L$(1):L$(4)=L$(2)
740 L$(7)=L$(5):L$(8)=L$(6)
750 DATA 1,3,24,17,17,6,11,11,11,7,17,17,17,17,17,
17,14,15,15,15,16,17,17,17,17,17,17
760 DATA 2,3,24,9,11,11,8,17,17,17,17,17,17,17,17,
9,11,11,8,17,17,17,17,17,17,17,17
770 DATA 5,6,12,4,12,13,13,12,17,17,2,3,17,17,17
780 DATA 6,6,12,12,12,5,17,17,17,17,0,1,17,17,17
880 REM -----------------------------
890 REM COULEURS
900 MODE 1:INK 0,13:INK 1,24:INK 2,6:INK 3,2:PAPER
 0:PEN 1:BORDER 13:PRINT
910 FOR I=1 TO 22
920 READ A,B:PAPER A:PEN B:PRINT STRING$(40,32);
930 P(I)=A:I(I)=B
940 NEXT I
950 DATA 0,1,0,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,3,1,2
,1,2,1
960 DATA 0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,1
970 FOR I=1 TO 7 STEP 4
980 D(I)=1:D(I+1)=26
990 D(I+2)=5:D(I+3)=21
1000 NEXT I
1010 PEN 1:PAPER 0
1020 LOCATE 7,2:PRINT CHR$(139)CHR$(139)"    "CHR$
(139)CHR$(139);
1030 PRINT "    "CHR$(139)CHR$(139)"    "CHR$(139)
CHR$(139)"    "CHR$(139)CHR$(139)
1040 RETURN
1080 REM -----------------------------
1090 REM PERDU
1100 FOR I=50 TO 500 STEP 5:SOUND 1,I,2:NEXT
1110 V=V-1
1120 IF V<0 THEN LINE INPUT A$:RUN
1130 LOCATE X,Y:PRINT" ":LOCATE 24,1:PRINT "VIES:"
;V
1140 GOTO 80
1180 REM -----------------------------
1190 REM GAGNE
1200 FOR I=500 TO 50 STEP-5:SOUND 1,I,2:NEXT
1210 T=T+1:P=P+10
1220 IF T=5 THEN GOSUB 1010:P=P+50:GOTO 70
1230 PEN 1:PAPER 0:LOCATE X-1,Y:PRINT "   "
1240 GOTO 80
```

Programme

Lignes 30 à 80 : initialisation

Cette première partie du programme appelle le sous-programme en 500 qui exécute les tâches suivantes:

1 – Définition des caractères graphiques (lignes 500 à 660).

2 – Définition du tableau L$ contenant les objets à déplacer sur l'écran et les pointeurs correspondants (tableau D. lignes 670 à 780).

3 – Affectation des codes couleurs pour le papier et l'encre à chaque ligne de l'écran (lignes 900 à 1040).

Les variables X et Y donnent la position de départ de la grenouille.

Lignes 90 à 190 : le jeu

Après l'affichage de la grenouille et la sauvegarde de ses coordonnées dans AX et AY, on place sur l'écran les chaînes de caractères du tableau L$ à partir des pointeurs D().

Si la grenouille est déjà sur la ligne 2 de l'écran (ligne 140), on va en 1200 pour le comptage des rochers.

Les lignes 150 à 180 modifient le tableau des pointeurs D pour le déplacement des objets.

Lignes 300 à 420 : déplacements

En fonction de la touche qui vient d'être appuyée, on modifie les variables X et Y. On place dans C la couleur qui correspond à ces nouvelles coordonnées.

Si C est égal à un blanc et si la grenouille est sur la rivière (ligne 360), on va en 1100 pour la phase « PERDU ». Idem quand C est différent d'un espace et que la grenouille se trouve sur la route (ligne 370).

Si la grenouille dépasse les limites de l'écran (en ligne 380), on va aussi en 1100.

La grenouille devant suivre le cours de l'eau, on la déplace à droite ou à gauche suivant qu'elle se trouve sur un tronc d'arbre ou un crocodile (lignes 400 et 410).

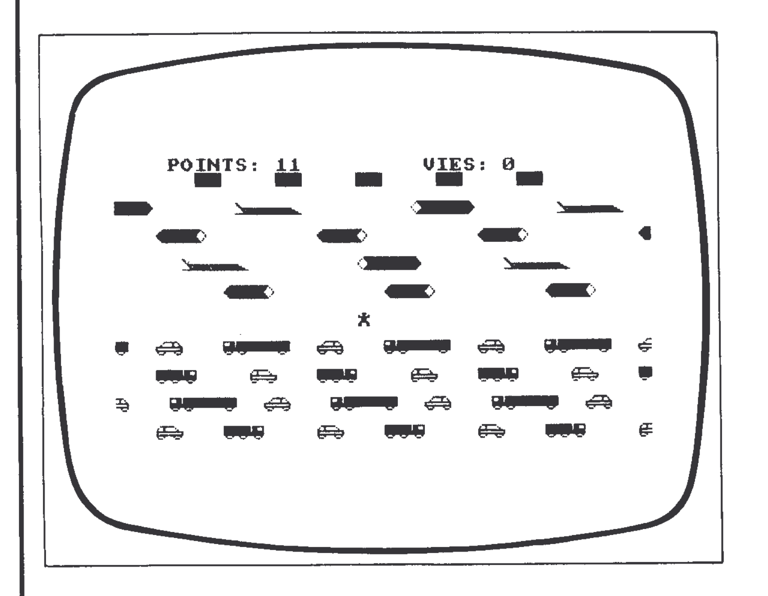
Lignes 1100 à 1240 : perdu et gagné

Si l'on a perdu une vie, on décrémente le compteur V, et l'on revient au jeu uniquement si V est encore supérieur à 0.

Si l'on a gagné, on incrémente le compteur de rochers T de 1 et le compteur de points de 10. Si T est inférieur à 5 on continue le jeu, sinon, on recommence une partie en incrémentant le compteur P de 50 points.

LISTE DES VARIABLES :

A	utilisée pour le déplacement des objets: A=2 lignes du haut, A=4 lignes du bas
A$	dernière touche appuyée
AX et **AY**	anciennes coordonnées de la grenouille
C	couleur sous la grenouille
D()	pointeurs dans les chaînes L$()
I	indice de boucle FOR/NEXT
I()	table des stylos
J	indice de boucle FOR/NEXT
L$()	chaînes d'objets
P	nombre de points
P()	table des papiers
T	nombre de rochers
V	nombre de vies
X et Y	coordonnées de la grenouille





CANADAIR

Qui n'a jamais rêvé d'être seul maître à bord d'un avion ? Voilà l'occasion d'exercer vos talents de pilote.

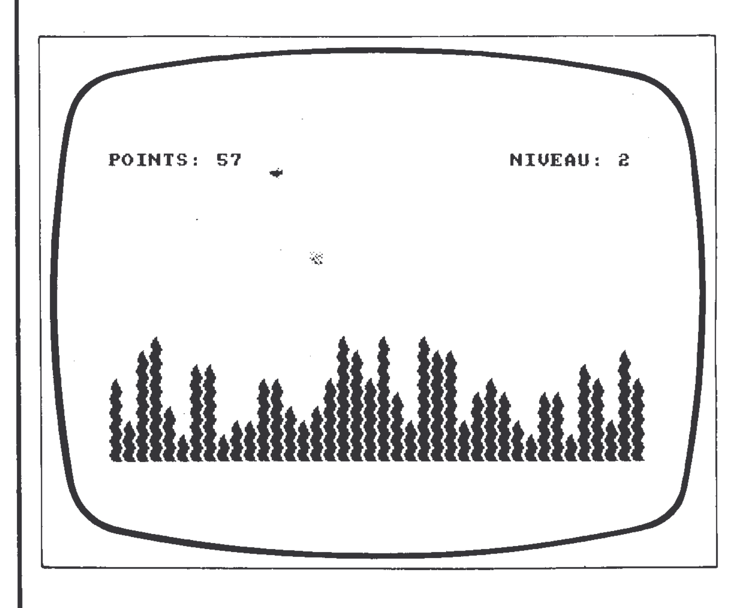
Un feu de forêt vient d'être annoncé et le canadair doit jeter son eau sur les flammes dessinées à l'écran.

Au premier tableau, le canadair va et vient, d'abord lentement, puis sa vitesse est doublée définitivement.

Le largage de l'eau se fait par la barre d'espace ainsi que les changements de tableaux quand toutes les flammes sont éteintes, et les remises en jeu.

Attention : aux tableaux suivants, les flammes montent de plus en plus haut et nécessitent une bonne réflexion stratégique pendant la descente.

Quand l'avion heurte une flamme, il s'écrase et repart à zéro.



```
10 REM CANADAIR
20 REM --------------------------------
30 SYMBOL AFTER 128:GOSUB 1000
40 MODE 1:INK 0,0:INK 1,6:INK 2,24:INK 3,2:PAPER 0
:PEN 1:BORDER 0
50 RANDOMIZE TIME:N=1:P=0.5
60 I=P
70 PEN 2:LOCATE 31,1:PRINT"NIVEAU:";N:PEN 1
180 REM --------------------------------
190 REM DESSIN DU JEU
200 FOR X=1 TO 40
210 A=INT(RND*N*4)
220 LOCATE X,21-A:PRINT CHR$(128)
230 FOR Y=22-A TO 22
240 LOCATE X,Y:PRINT CHR$(129)
250 NEXT Y
260 NEXT X
270 X=1
280 Y=1
380 REM --------------------------------
390 REM LE JEU
400 PEN 2:LOCATE X,Y:PRINT " "
410 IF X>=40 THEN I=-P
420 IF X<=1 THEN I=P:Y=Y+1
430 IF Y=4 THEN P=1
440 IF Y=22 THEN 900
450 X=X+I
460 IF TEST(X*16-8,408-Y*16)>0 THEN 800
470 IF I>0 THEN LOCATE X,Y:PRINT CHR$(131):GOTO 49
0
480 LOCATE X,Y:PRINT CHR$(132)
490 FOR K=1 TO 30:NEXT:IF F=1 THEN 540
500 IF INKEY$="" THEN 400
510 F=1
520 L=Y+1
530 C=X
540 LOCATE C,L:PRINT " "
550 L=L+1
560 IF L>=23 THEN F=0:GOTO 400
570 IF TEST(C*16-8,408-L*16)>0 THEN 600
580 PEN 3:LOCATE C,L:PRINT CHR$(130)
590 GOTO 400
600 F=0:T=T+1:PEN 2:LOCATE 1,1:PRINT "POINTS:";T
610 FOR J=L TO L+2
620 IF J<23 THEN LOCATE C,J:PRINT " "
630 NEXT J:L=L+2
640 IF L>=22 THEN 400
650 PEN 1:LOCATE C,L:PRINT CHR$(128)
660 GOTO 400
780 REM --------------------------------
790 REM PERDU
```

```
800 IF X>=40 THEN X=39
810 FOR I=Y TO 21
820 LOCATE X,I-1:PRINT"   "
830 LOCATE X,I:PRINT"**":LOCATE X,I+1:PRINT CHR$(1
31)"*"
840 NEXT I
850 IF INKEY$="" THEN 850
860 RUN
880 REM ------------------------------
890 REM GAGNE
900 FOR I=500 TO 50 STEP -5:SOUND 1,I,2:NEXT
910 IF INKEY$="" THEN 910
920 N=N+1:IF N=5 THEN N=4
930 GOTO 60
980 REM ------------------------------
990 REM GRAPHIQUES
1000 SYMBOL 128,8,24,56,60,124,126,126,63
1010 SYMBOL 129,63,126,126,252,252,126,126,63
1020 SYMBOL 130,162,73,22,169,100,18,91,36
1030 SYMBOL 131,32,144,255,255,60,16,32,0
1040 SYMBOL 132,4,9,255,255,60,8,4,0
1050 RETURN
```

LISTE DES VARIABLES :

A	nombre aléatoire : hauteur des flammes
C et **L**	coordonnées de l'eau
F	drapeau: si F=1 l'eau est en cours de descente
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
N	niveau du jeu
P	incrément de base
T	nombre de points
X et Y	coordonnées de l'avion

Programme

Lignes 30 à 70 : initialisation

Après la définition des graphiques par le sous-programme en 1000, on initialise les diverses variables (voir liste des variables).

Lignes 200 à 280 : dessin du jeu

La variable N donnant le niveau du jeu, on affiche à partir du bas de l'écran des flammes dont la hauteur est déterminée par un nombre aléatoire fonction de N.

Les variables X et Y définissent la position du canadair.

Lignes 400 à 660 : le jeu

Si le canadair arrive à droite de l'écran (X>=40), on change de direction en modifiant la valeur de l'incrément I (ligne 410).

On effectue l'opération inverse s'il arrive à gauche (X<=1 en ligne 420) et l'on descend d'une ligne.

Quand on arrive sur la quatrième ligne (Y=4), on change l'incrément de base pour aller plus vite (de 0.5 à 1).

La 22e ligne étant la dernière, on incrémente le niveau du jeu N, et l'on commence une nouvelle partie (ligne 440).

Une fois la position du canadair modifiée (ligne 450), on teste sur l'écran le caractère qui s'y trouve. Si celui-ci est différent de 0 (une flamme), on va en ligne 800 pour la fin du jeu.

Suivant la valeur de I (positive ou négative), on affiche un canadair vers la gauche ou vers la droite (lignes 470 et 480).

Le drapeau F indique que de l'eau est déjà en train de descendre. On va alors directement en ligne 540.

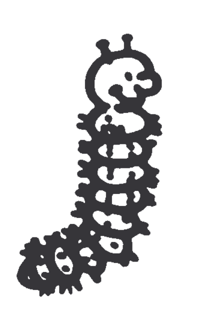
Dans l'autre cas, on teste le clavier pour savoir si l'on doit jeter de l'eau ou non.

Les lignes 520 à 590 affichent sur l'écran l'eau qui descend. Dès que celle-ci touche la flamme (testée en ligne 570), on en efface une partie (lignes 600 à 660).

Lignes 800 à 930 : perdu et gagné

Si le canadair a touché une flamme, on le fait descendre jusqu'au bas de l'écran.

Mais si c'est gagné, on incrémente le niveau du jeu N avant de reprendre une autre partie.



CHENILLE

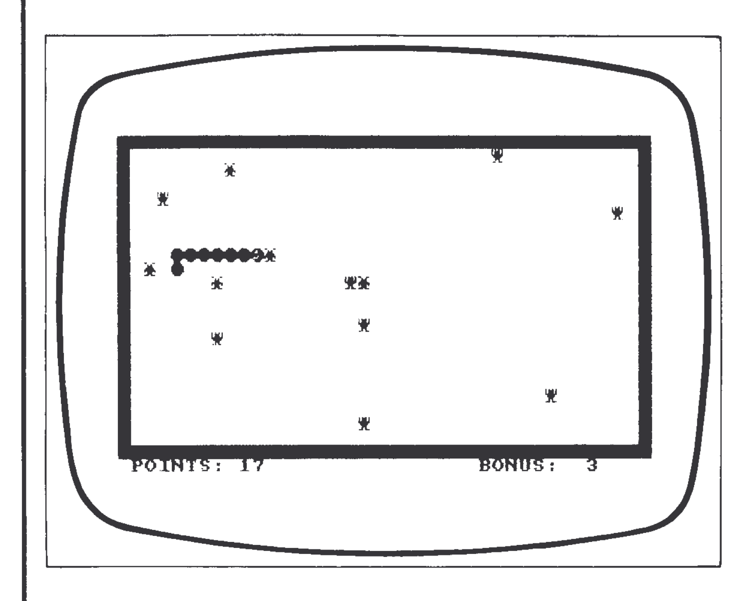
Vous devez constituer une chenille en avalant tous les éléments apparaissant sur l'écran. Ils sont de deux sortes : les bons qui donnent un bonus (ceux qui sont rouges), et les autres qui en font perdre un (ceux qui sont violets).

Le compte des bonus s'affiche en bas à droite de l'écran: il faut bien sûr que ce compteur soit supérieur à 0 pour prétendre manger un « anti-bonus ».

La chenille s'allonge à chaque élément qu'elle avale. Elle se déplace toute seule, le joueur devant modifier sa trajectoire avec les touches P et Q.

Mais attention à ne pas heurter le bord du tableau ou enrouler entièrement la chenille sur elle-même !

A chaque tableau réussi succède un nouveau tableau comportant un nombre de plus en plus important d'éléments. Ne vous découragez surtout pas au premier tableau : la dextérité s'acquiert très vite.



```
10 REM CHENILLE
20 REM -----------------------------
30 SYMBOL AFTER 128:GOSUB 1000
40 DIM Y(201),X(201)
50 E=0:TT=0
60 RANDOMIZE TIME
80 REM -----------------------------
90 REM DESSIN DU JEU
100 P=1:Q=1:T=0:E=E+10:IF E>200 THEN E=200
110 DX=0:DY=-1:X(1)=20:Y(1)=12:B=0
120 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,8:PAPER
1:PEN 1:BORDER 1
130 FOR I=1 TO 40
140 LOCATE I,1:PRINT " ":LOCATE I,23:PRINT " "
150 IF I<23 THEN LOCATE 1,I:PRINT " ":LOCATE 40,I:
PRINT " "
160 NEXT I:PAPER 0
170 LOCATE X(1),Y(1):PRINT CHR$(128)
180 FOR I=1 TO E
190 Y=INT(RND*21)+2:X=INT(RND*38)+2
200 IF TEST(X*16-8,408-Y*16)>0 THEN 190
210 IF I>E/2 THEN PEN 2:LOCATE X,Y:PRINT CHR$(130)
 ELSE PEN 3:LOCATE X,Y:PRINT CHR$(131)
220 NEXT I
230 PEN 1:LOCATE 2,24:PRINT"POINTS:";TT;TAB(28);"B
ONUS: 0"
240 IF INKEY$="" THEN 240
280 REM -----------------------------
290 REM LE JEU
300 D$=INKEY$:IF D$="" THEN 380
310 IF DY=0 THEN 350
320 IF D$="Q" THEN DX=DY:GOTO 340
330 DX=-DY
340 DY=0:GOTO 380
350 IF D$="P" THEN DY=DX:GOTO 370
360 DY=-DX
370 DX=0
380 LOCATE X(P),Y(P):PRINT CHR$(129)
390 AP=P:P=P+1:IF P=202 THEN P=1
400 X(P)=X(AP)+DX:Y(P)=Y(AP)+DY
410 A=TEST(X(P)*16-8,408-Y(P)*16):IF A<>0 THEN 500
420 LOCATE X(Q),Y(Q):PRINT " "
430 Q=Q+1:IF Q=202 THEN Q=1
440 LOCATE X(P),Y(P):PRINT CHR$(128)
450 GOTO 300
480 REM -----------------------------
490 REM ELEMENT RENCONTRE ?
500 IF A=1 THEN 800
510 IF A=2 THEN B=B+1:GOTO 550
520 IF B=0 THEN 800
530 B=B-1
```

```
550 T=T+1:TT=TT+1
560 LOCATE 9,24:PRINT TT:LOCATE 35,24:PRINT B;" "
570 IF T=E THEN 100
580 GOTO 440
780 REM ------------------------------
790 REM FIN
800 LOCATE 9,24:PRINT TT:LOCATE 35,24:PRINT B;" "
810 LOCATE X(Q),Y(Q):PRINT " "
820 Q=Q+1:IF Q=202 THEN Q=1
830 IF Q<>P THEN 810
840 IF INKEY$<>CHR$(13) THEN 840
850 RUN
980 REM ------------------------------
990 REM GRAPHIQUES
1000 SYMBOL 128,60,110,207,255,247,143,126,60
1010 SYMBOL 129,60,126,255,255,255,255,126,60
1020 SYMBOL 130,198,40,56,124,254,186,56,238
1030 SYMBOL 131,170,186,186,254,124,56,40,238
1040 RETURN
```

LISTE DES VARIABLES :

A	couleur du point rencontré par la chenille
AP	ancienne valeur de P
B	nombre de bonus
D$	dernière touche enfoncée
DX et **DY**	sens du déplacement en X et en Y de la chenille
E	nombre d'éléments à éliminer
I	indice de boucle FOR/NEXT
P	pointeur de la tête de la chenille dans X() et Y()
Q	idem pour la queue de la chenille
T	nombre de points du tableau
TT	total des points de la partie
X et **Y**	nombres aléatoires : coordonnées des bonus et anti-bonus
X() et **Y()**	coordonnées de chaque élément de la chenille

Programme

Lignes 30 à 60 : initialisation

Après la définition des caractères graphiques par le sous-programme en 1000, on dimensionne deux tableaux X et Y pour connaître à tout moment la position des éléments qui composent la chenille.

Le compteur E donne le nombre d'éléments à éliminer, tandis que TT compte le nombre de points de toutes les parties.

La ligne 60 initialise les nombres aléatoires. On utilise pour cela la variable système TIME. Celle-ci étant toujours différente, on ne peut connaître sa valeur à l'avance.

Lignes 100 à 240 : dessin du jeu

Après l'effacement de l'écran et le dessin du cadre (lignes 120 à 160), on affiche aléatoirement sur l'écran les deux sortes d'éléments (caractères graphiques 130 et 131).

Lignes 300 à 450 : le jeu

Si une touche est appuyée, on modifie en conséquence les incréments de direction DX et DY (lignes 300 à 370). Puis l'on affiche, à la place de la tête de la chenille pointée par la variable P, un élément du corps.

Après l'incrément de P, on modifie le contenu de X(P) et Y(P) avec les incréments DX et DY (ligne 400).

Si un élément est rencontré (testé par TEST en ligne 410), on va en 500 pour identification.

Dans le cas contraire, on efface le dernier élément de la chenille pointé par la variable Q. Cette dernière variable est incrémentée dans la limite des tableaux X et Y.

Pour terminer cette partie du programme, on affiche la tête de la chenille aux coordonnées pointées par P.

Lignes 500 à 580 : élément rencontré

Si l'élément rencontré est le bord ou la chenille (A=1), on va directement en ligne 800 pour la fin du jeu.

Par contre, s'il s'agit d'un bonus, on incrémente le compteur B. Puisqu'il ne reste plus que les « anti-bonus » à tester, on va en ligne 800 pour la fin du jeu si le compteur B est à zéro.

Dans l'autre cas, on décrémente la variable B.

La variable T indique le nombre d'éléments déjà éliminés. On la compare donc à E pour connaître le moment de fin d'un tableau.Si ces deux variables sont égales, on va en ligne 100 où le compteur E est incrémenté de 10 pour le nouveau tableau (dans la limite de 200 éléments).



INCENDIE

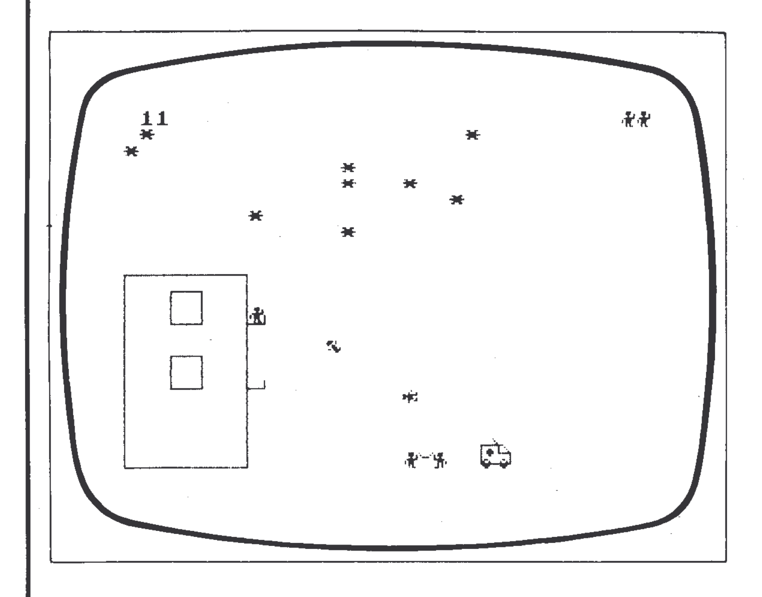
Après le lancement du programme, un décor se dessine instantanément sur l'écran.

Dans le ciel, des étoiles et, au milieu de l'écran, une maison, une ambulance et deux pompiers. Ces derniers doivent rattraper tous les personnages (1, puis 2, puis 3, jusqu'à 10) qui tombent de la maison en feu, et les mettre dans l'ambulance.

Vous les manœuvrez aves les touches P et Q.

Vous gagnez un point quand l'un des personnages atteint l'ambulance. Par contre vous perdez une des trois vies qui vous sont données au départ quand l'un des bonhommes s'écrase au sol. De plus, au bout de 50 points, vous gagnez une nouvelle vie.

De quoi peut-être faire naître une vocation !



```
10 REM INCENDIE
20 REM ------------------------------
30 SYMBOL AFTER 128:DIM X(15),Y(15),C(15):GOSUB 11
00
40 RANDOMIZE TIME
50 MODE 1:INK 0,3:INK 1,0:INK 2,15:INK 3,2:PAPER 0
:PEN 1:BORDER 3:GOSUB 900
60 N=1:P=1:S=2:T=0
70 IF INKEY$="" THEN 70
180 REM -----------------------------
190 REM LE JEU
200 GOSUB 800
210 PEN 1:LOCATE 11,22:PRINT A$(P)
220 FOR I=1 TO N
230 P(I)=0
240 NEXT I
250 FOR I=1 TO N
260 IF I=1 THEN 290
270 IF P(I-1)=4 THEN LOCATE 9,13+INT(RND+0.5)*4:PR
INT CHR$(132)
280 IF P(I-1)<6 THEN 440
290 IF P(I)>14 THEN 440
300 A$=INKEY$
310 P=P+(A$="P")*(P<3)-(A$="Q")*(P>1)
320 LOCATE 11,22:PRINT A$(P)
330 IF P(I)=0 THEN LOCATE 9,13:PRINT " ":LOCATE 9,
17:PRINT " ":GOTO 350
340 LOCATE X(P(I)),Y(P(I)):PRINT " "
350 P(I)=P(I)+1
360 IF P(I)<>15 THEN 390
370 T=T+1:IF T/50=INT(T/50) THEN S=S+1:GOSUB 800
380 PEN 2:LOCATE 1,1:PRINT T;CHR$(7);:PEN 1:GOTO 4
40
390 LOCATE X(P(I)),Y(P(I)):PRINT CHR$(128+C(P(I)))
400 FOR J=1 TO 11-N
410 IF J=1 THEN A$=INKEY$:P=P+(A$="P")*(P<3)-(A$="
Q")*(P>1):LOCATE 11,22:PRINT A$(P)
420 NEXT
430 IF Y(P(I))=21 THEN SOUND 1,500,2:IF TEST(X(P(I
))*16-8,58)=0 THEN 600
440 NEXT I
450 FOR J=1 TO N
460 IF P(J)<15 THEN 250
470 NEXT J
480 N=N+1:IF N>10 THEN N=10
490 GOTO 200
580 REM -----------------------------
590 REM PERDU
600 LOCATE X(P(I)),21:PRINT " "
610 LOCATE X(P(I)),22:PRINT CHR$(131)
620 SOUND 1,426,2:SOUND 1,478,2:SOUND 1,379,2
630 S=S-1:GOSUB 800
640 P(I)=16
```

```
650 IF S>-1 THEN 440
660 FOR I=50 TO 500 STEP 5:SOUND 1,I,2:NEXT
670 IF INKEY$<>CHR$(13) THEN 670 ELSE 50
780 REM ------------------------------
790 REM AFFICHE LES VIES
800 PEN 3:LOCATE 33,1:PRINT"        "
810 FOR J=1 TO S
820 LOCATE 32+J,1:PRINT CHR$(132);"  "
830 NEXT J
840 PEN 1:RETURN
880 REM ------------------------------
890 REM DESSIN DU JEU
900 PLOT 0,238:DRAW 126,238:DRAW 126,48:DRAW 0,48:
DRAW 0,238
910 PLOT 126,190:DRAW 144,190:DRAW 144,198
920 PLOT 126,126:DRAW 144,126:DRAW 144,134
930 PLOT 48,222:DRAW 80,222:DRAW 80,190:DRAW 48,19
0:DRAW 48,222
940 PLOT 48,158:DRAW 80,158:DRAW 80,126:DRAW 48,12
6:DRAW 48,158
950 LOCATE 24,21:PRINT CHR$(135)CHR$(136)
960 LOCATE 24,22:PRINT CHR$(137)CHR$(138)
970 FOR I=1 TO 10
980 LOCATE 1+RND*40,1+RND*8:PRINT "*"
990 NEXT I
1000 RETURN
1080 REM ------------------------------
1090 REM GRAPHIQUES
1100 SYMBOL 128,56,186,146,126,56,40,38,48
1110 SYMBOL 129,2,116,228,88,29,111,6,0
1120 SYMBOL 130,0,8,17,223,248,221,19,16
1130 SYMBOL 131,54,20,20,28,127,73,93,93
1140 SYMBOL 132,28,29,9,63,92,92,20,22
1150 SYMBOL 133,129,66,60,0,0,0,0,0
1160 SYMBOL 134,56,184,144,252,58,42,40,232
1170 SYMBOL 135,0,2,1,0,255,128,140,158
1180 SYMBOL 136,0,0,0,128,240,72,68,66
1190 SYMBOL 137,158,140,128,176,200,255,120,48
1200 SYMBOL 138,127,1,1,25,37,255,60,24
1210 A$(1)=CHR$(132)+CHR$(133)+CHR$(134)+"
"
1220 A$(2)="    "+CHR$(132)+CHR$(133)+CHR$(134)+"
    "
1230 A$(3)="         "+CHR$(132)+CHR$(133)+CHR$(134
)
1240 FOR I=1 TO 14
1250 READ X(I),Y(I),C(I)
1260 NEXT I
1270 RETURN
1280 DATA 10,15,1,11,18,2,12,21,3,13,18,2
1290 DATA 14,15,1,15,18,2,16,21,3,17,18,2
1300 DATA 18,15,1,19,18,2,20,21,3,21,18,2,22,15,3,
23,18,2
```

Programme

Lignes 30 à 70 : initialisation

Pour connaître la position des personnages et les caractères graphiques qui s'y rapportent, on dimensionne trois tableaux X, Y et C. Ceux-ci sont remplis par le sous-programme 1100. C'est aussi ce sous-programme qui définit les caractères graphiques.

Lignes 200 à 490 : le jeu

Le sous-programme en 800 affiche des personnages en haut de l'écran pour indiquer au joueur le nombre de vies restantes.

Chaque personnage a une position bien déterminée contenue dans le tableau P. Celui-ci est donc remis à zéro aux lignes 220 à 240.

La variable N donnant le nombre de personnages à récupérer, on effectue cette boucle de 1 à N le nombre de fois qu'il faut pour qu'ils soient tous dans l'ambulance.

Les personnages apparaissent au balcon uniquement si le précédent a déjà couvert quatre positions (ligne 270), et ils sautent uniquement quand le précédent en est à six positions (ligne 280).

Le caractère saisi par INKEY$ en ligne 300 permet de déterminer la position du drap (variable P). On utilise pour cela une expression logique pour modifier la valeur de P (ligne 310). Pour permettre de bouger plus rapidement le drap, on répète cette ligne en 410.

Quand un personnage atteint la position 15 (l'ambulance), on incrémente le compteur de points T (ligne 370). Si T est divisible par 50, on ajoute une vie (compteur S).

Dans le cas où un personnage arrive sur le sol, on teste par la fonction TEST la présence du drap (ligne 430). Si la couleur trouvée à cet endroit est 0 (un blanc), on va à la ligne 600 pour la fin du jeu.

Les lignes 450 à 470 testent si tous les personnages sont dans l'ambulance. Si ce n'est pas le cas, on revient en ligne 250. Tous les personnages étant arrivés à bon port, on incrémente le compteur N pour en agrandir la famille au prochain tour (ligne 480).

LISTE DES VARIABLES :

A$()	dessin du drap pour les trois positions
C()	pointeurs des caractères graphiques de chaque position des personnages
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
N	nombre de sauvetages à effectuer
P	position du drap
P()	position de chaque personnage
S	nombre de vies
T	nombre de points
X() et **Y()**	toutes les coordonnées possibles des personnages.



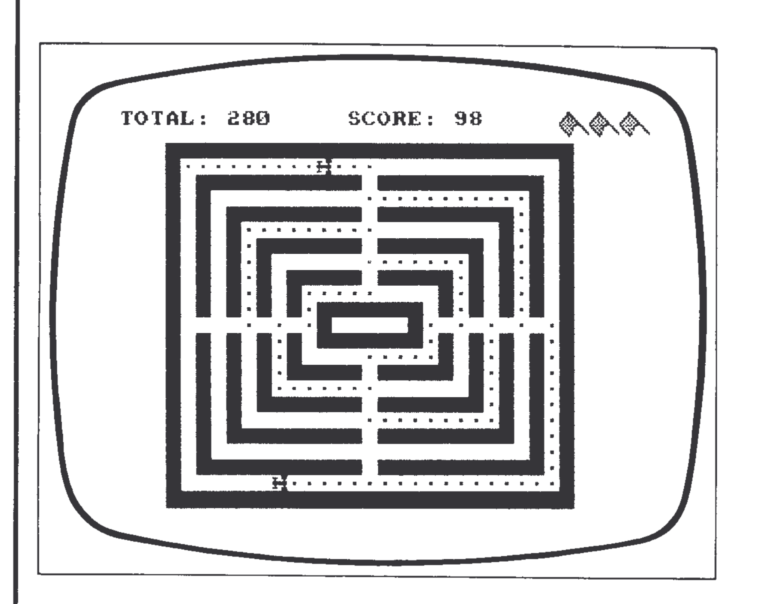
COURSE AUTOMOBILE

Eviterez-vous la voiture infernale de l'ordinateur qui roule en sens inverse, tout en passant sur chaque portion du circuit ?

Deux commandes sont nécessaires à votre déplacement ; la touche P pour vous diriger vers le centre et la touche Q pour aller vers l'extérieur.

Dès le début de la course, trois parties gratuites vous sont allouées. Chaque déplacement donne un point. Les points sont affichés en haut de l'écran à chaque déplacement vers le centre ou vers l'extérieur. Si vous passez sur toutes les portions du circuit, une nouvelle piste s'affiche avec le total général et le jeu reprend de plus belle.

Bonne chance... et pas d'imprudence !



```
10 REM COURSE AUTOMOBILE
20 REM ------------------------------
30 SYMBOL AFTER 128:GOSUB 1300
40 DEFINT A-Z
50 GOSUB 1100
60 X(0)=1:X(2)=-1:Y(1)=1:Y(3)=-1
70 PG=3
80 FOR I=1 TO PG
90 LOCATE 31+I*2,1:PRINT CHR$(132)CHR$(133)"   ":L
OCATE 31+I*2,2:PRINT CHR$(134)CHR$(135)"   "
100 NEXT I
110 S=0:SO=3
120 X=8:Y=4:XO=32:YO=23
130 L=5:LO=5:D=0:A=32
180 REM ------------------------------
190 REM LE JEU
200 PEN 2:LOCATE X,Y:PRINT CHR$(136):PEN 1
210 X=X+X(S):Y=Y+Y(S)
220 IF TEST(X*16-8,406-Y*16)=0 THEN G=G+1:IF G=279
 THEN 800
230 LOCATE X,Y:PRINT CHR$(S+128)
240 PEN 2:LOCATE XO,YO:PRINT CHR$(A):PEN 1:XO=XO+X
(SO):YO=YO+Y(SO)
250 A=TEST(XO*16-8,406-YO*16):LOCATE XO,YO:PRINT C
HR$(SO+128):IF A=2 THEN A=136 ELSE A=32
260 IF L<LO THEN D=3
270 IF L>LO THEN D=1
280 IF D<>0 AND TEST((XO+X((SO+D) MOD 4))*16-8,406
-(YO+Y((SO+D) MOD 4))*16)=0 AND TEST((XO+X(SO))*16
-8,406-(YO+Y(SO))*16)<>3 THEN 700
290 Z=TEST((X+X(S))*16-8,406-(Y+Y(S))*16):IF Z=1 T
HEN 800
300 IF Z=0 OR Z=2 THEN 320
310 S=(S+1) MOD 4
320 B$=INKEY$:IF B$<>"" THEN C$=B$
330 IF C$="P" THEN 500
340 IF C$="Q" THEN 600
350 Z=TEST((XO+X(SO))*16-8,406-(YO+Y(SO))*16):IF Z
=0 OR Z=2 THEN 200
360 SO=(SO+3) MOD 4
370 GOTO 200
480 REM ------------------------------
490 REM DEPLACEMENT CENTRE
500 IF TEST((X+X((S+1) MOD 4))*16-8,406-(Y+Y((S+1)
 MOD 4))*16)<>0 THEN 350
510 SOUND 1,200,1:PEN 2:LOCATE X,Y:PRINT CHR$(136)
:PEN 1
520 X=X+X((S+1) MOD 4)+X((S+1) MOD 4):Y=Y+Y((S+1)
MOD 4)+Y((S+1) MOD 4)
530 L=L-1:IF L<1 THEN L=1
540 IF TEST(X*16-8,406-Y*16)=0 THEN G=G+1
```

```
550 LOCATE 25,1:PRINT G
560 C$="":GOTO 350
580 REM ------------------------------
590 REM DEPLACEMENT EXTERIEUR
600 IF TEST((X+X((S+3) MOD 4))*16-8,406-(Y+Y((S+3)
 MOD 4))*16)<>0 THEN 350
610 SOUND 1,50,1:PEN 2:LOCATE X,Y:PRINT CHR$(136):
PEN 1
620 X=X+X((S+3) MOD 4)+X((S+3) MOD 4):Y=Y+Y((S+3)
MOD 4)+Y((S+3) MOD 4)
630 L=L+1
640 IF TEST(X*16-8,406-Y*16)=0 THEN G=G+1
650 LOCATE 25,1:PRINT G
660 C$="":GOTO 350
680 REM ------------------------------
690 REM DEPLACEMENT AMSTRAD
700 PEN 2:LOCATE XO,YO:PRINT CHR$(A):PEN 1
710 XO=XO+X((SO+D) MOD 4)+X((SO+D) MOD 4):YO=YO+Y(
(SO+D) MOD 4)+Y((SO+D) MOD 4)
720 IF D=1 THEN LO=LO+1 ELSE LO=LO-1
730 D=0:A=TEST(XO*16-8,406-YO*16):IF A=2 THEN A=13
6 ELSE A=32
740 GOTO 290
780 REM ------------------------------
790 REM FIN DE PARTIE
800 PEN 2:LOCATE XO,YO:PRINT CHR$(A):LOCATE X,Y:PR
INT CHR$(136):PEN 1
810 IF G=279 THEN G=280:PG=PG+1:FOR I=500 TO 50 ST
EP-5:SOUND 1,I,2:NEXT ELSE FOR I=50 TO 500 STEP 5:
SOUND 1,I,2:NEXT
820 PG=PG-1:TP=TP+G:G=0:LOCATE 4,1:PRINT"TOTAL:";T
P
830  FOR I=3 TO 12 STEP 2:FOR J=0 TO 30-I*2
840  LOCATE I+5+J,I+1:PRINT" ":LOCATE I+5+J,27-I:P
RINT" ":IF J<26-I*2 THEN LOCATE I+5,I+J+1:PRINT" "
:LOCATE 35-I,I+J+1:PRINT" "
850  NEXT:NEXT
860 IF PG=0 THEN 1000
870 GOTO 80
980 REM ------------------------------
990 REM FIN DU JEU
1000 LOCATE 33,1:PRINT "  ":LOCATE 33,2:PRINT "  "
1010 LOCATE 16,14:PRINT"GAME OVER"
1020 IF INKEY$<>CHR$(13) THEN 1020
1030 RUN
1080 REM ------------------------------
1090 REM DESSIN DU JEU
1100 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPE
R 3:PEN 1:BORDER 1
1110 FOR I=2 TO 13 STEP 2:FOR J=0 TO 30-I*2
```

```
1120 LOCATE I+5+J,I+1:PRINT" ":LOCATE I+5+J,27-I:P
RINT" ":IF J<26-I*2 THEN LOCATE I+5,I+J+1:PRINT" "
:LOCATE 35-I,I+J+1:PRINT" "
1130 NEXT:NEXT
1140 PAPER 0:LOCATE 9,14:PRINT"       ":LOCATE 25,
14:PRINT"       "
1150 FOR I=5 TO 12
1160 LOCATE 20,I:PRINT " ":LOCATE 20,I+12:PRINT "
"
1170 NEXT I
1180 LOCATE 19,1:PRINT"SCORE:"
1190 RETURN
1280 REM ------------------------------
1290 REM GRAPHIQUES
1300 SYMBOL 128,224,231,66,255,255,66,231,224
1310 SYMBOL 129,219,255,219,24,24,90,126,90
1320 SYMBOL 130,7,231,66,255,255,66,231,7
1330 SYMBOL 131,90,126,90,24,24,219,255,219
1340 SYMBOL 132,12,14,27,53,106,213,170,213
1350 SYMBOL 133,0,0,0,128,192,96,240,152
1360 SYMBOL 134,107,53,26,14,4,0,0,0
1370 SYMBOL 135,12,6,3,1,0,0,0,0
1380 SYMBOL 136,0,0,0,24,24,0,0,0
1390 RETURN
```

Programme

Lignes 30 à 380 : le jeu

Après l'initialisation des graphiques et du dessin du jeu (sous-programme en 1300 et 1100), on affiche en haut de l'écran le nombre de parties gratuites.

La ligne 200 place un point sur la position du joueur.

Si la nouvelle position comporte un espace (TEST=0), on incrémente le compteur de points G. Si G=279 on va en 800 pour une nouvelle partie.

On répète la même séquence pour la voiture de l'ordinateur.

Les variables L et LO donnant le numéro de piste (L=5 pour la piste extérieure et L=1 pour la piste intérieure) on place dans D le sens du déplacement de l'ordinateur (D=3 pour l'intérieur et D=1 pour l'extérieur). Dans le cas où D est différente de zéro et qu'une case blanche est adjacente, on va en 700 pour le déplacement d'Amstrad.

Si la voiture du joueur rencontre celle de l'ordinateur (ligne 290), on saute à la fin de partie (ligne 800).

La touche appuyée (stockée dans B$) est gardée en mémoire dans la variable C$ jusqu'au prochain carrefour. Les lignes 330 et 340 orientent alors sur les parties concernées du programme.

Lignes 500 à 560 : déplacement centre

La touche P étant appuyée, on déplace la voiture du joueur vers le centre dans le cas où il existe une case blanche adjacente (TEST=0).

Puis l'on incrémente les variables X,Y et l'on décrémente le compteur L pour que l'ordinateur sache sur quelle piste on se trouve.

La ligne 550 affiche le nombre de points acquis.

Lignes 600 à 660 : déplacement extérieur

Identiques aux lignes 500 à 560 mais pour l'extérieur.

Lignes 700 à 740 : déplacement d'Amstrad

La variable D (sens du déplacement de la voiture du joueur) étant différente de zéro, on déplace la voiture d'Amstrad vers l'extérieur (si D=1) ou vers l'intérieur (si D=3).

Puis l'on incrémente les variables XO, YO et le compteur LO pour que l'ordinateur sache sur quelle piste il se trouve.

Lignes 800 à 870 : fin de partie

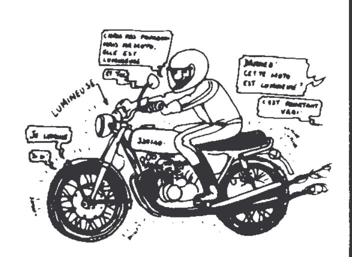
Si la voiture du joueur a rencontré celle de l'ordinateur ou si toutes les portions du circuit ont été couvertes, on affiche le score total et l'on efface les points pour une nouvelle partie.

Lignes 1000 à 1030 : fin du jeu

Comme il ne reste plus aucune partie gratuite (variable PG), on affiche un nouveau jeu dès que la touche ENTER est enfoncée.

LISTE DES VARIABLES :

A	contient le caractère qui effacera la voiture d'Amstrad au prochain déplacement
B$	dernière touche enfoncée
D	différence entre le joueur et Amstrad
G	nombre de points dans une partie
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
L	numéro de la piste du joueur
LO	numéro de la piste d'Amstrad
PG	nombre de parties gratuites
S	sens du déplacement du joueur
SO	sens du déplacement d'Amstrad
TP	total des points
X et **Y**	coordonnées de la voiture du joueur
XO et **YO**	coordonnées de la voiture d'Amstrad
X() et **Y()**	nombre de caractères de déplacement dans les 4 directions
Z	couleur devant la voiture du joueur ou d'Amstrad



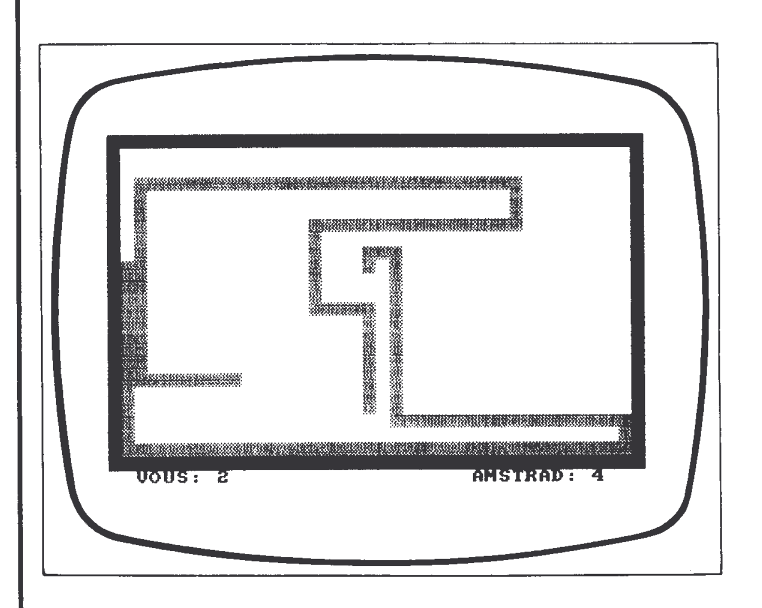
MOTOS LUMINEUSES

Votre moto déjouera-t-elle les pièges tendus par celle d'Amstrad ?

Les motos laissent une trace lumineuse derrière elles. Il faut donc éviter non seulement la trace de l'adversaire, mais aussi celle de votre propre moto. Celle-ci se dirige avec les touches P et Q du clavier.

Vous aurez besoin de beaucoup de dextérité pour enfermer la moto de l'ordinateur de manière à ce qu'elle revienne sur elle-même ou touche le bord de l'écran.

A chaque fin de partie, les scores du joueur et d'Amstrad s'affichent en bas de l'écran, et le jeu reprend pour mettre une fois de plus vos réflexes à dure épreuve.



```
10 REM MOTOS LUMINEUSES
20 REM ------------------------------
30 SYMBOL AFTER 128
40 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
1:BORDER 1:GOSUB 700:PAPER 0
50 RANDOMIZE TIME
60 X(0)=-1:X(2)=1:Y(1)=1:Y(3)=-1
70 X1=20:Y1=21:D1=3:X2=20:Y2=11:D2=INT(RND*4)
80 N=0:N1=0:N2=0
180 REM ----------------------------
190 REM LE JEU
200 A$=INKEY$
210 IF A$="Q" THEN D1=(D1+1)MOD 4
220 IF A$="P" THEN D1=(D1+3)MOD 4
230 X1=X1+X(D1):Y1=Y1+Y(D1)
240 IF TEST(X1*16-8,408-Y1*16)<>0 THEN 610
250 PEN 2:LOCATE X1,Y1:PRINT CHR$(129)
260 IF TEST((X2+X(D2))*16-8,408-(Y2+Y(D2))*16)<>0
OR N>N2 THEN GOSUB 400 ELSE N1=RND*8
270 X2=X2+X(D2):Y2=Y2+Y(D2)
280 PEN 3:LOCATE X2,Y2:PRINT CHR$(129)
290 N=N+INT(RND*2):GOTO 200
380 REM ----------------------------
390 REM MOUVEMENTS DE L'ORDINATEUR
400 N=0:N2=N2+N1
410 IF X(D2)=0 THEN 470
420 IF Y<10 AND N1<6 THEN 440
430 IF TEST(X2*16-8,408-(Y2-1)*16)=0 THEN D2=3:GOT
O 520
440 IF TEST(X2*16-8,408-(Y2+1)*16)=0 THEN D2=1:GOT
O 520
450 IF TEST(X2*16-8,408-(Y2-1)*16)=0 THEN D2=3:GOT
O 520
460 GOTO 510
470 IF X>30 AND N1<6 THEN 490
480 IF TEST((X2+1)*16-8,408-Y2*16)=0 THEN D2=2:GOT
O 520
490 IF TEST((X2-1)*16-8,408-Y2*16)=0 THEN D2=0:GOT
O 520
500 IF TEST((X2+1)*16-8,408-Y2*16)=0 THEN D2=2:GOT
O 520
510 IF TEST((X2+X(D2))*16-8,408-(Y2+Y(D2))*16)<>0
THEN 600
520 RETURN
580 REM ----------------------------
590 REM FIN DE PARTIE
600 T1=T1+1:GOTO 620
610 T2=T2+1
620 PEN 3:LOCATE 1,25:PRINT "  VOUS:";T1;TAB(28);"
AMSTRAD:";T2;
630 IF INKEY$<>CHR$(13) THEN 630
```

```
640 GOTO 40
680 REM --------------------------------
690 REM DESSIN DU JEU
700 SYMBOL 128,136,17,34,68,136,17,34,68
710 SYMBOL 129,85,170,85,170,85,170,85,170
720 FOR I=1 TO 40
730 LOCATE I,1:PRINT " ":LOCATE I,24:PRINT " "
740 IF I<25 THEN LOCATE 1,I:PRINT " ":LOCATE 40,I:
PRINT " "
750 NEXT I
760 PAPER 0:PEN 3:LOCATE 1,25:PRINT "  VOUS:";T1;T
AB(28);"AMSTRAD:";T2;
770 RETURN
```

LISTE DES VARIABLES :

A$	dernière touche enfoncée
D1 et **D2**	direction prise par le joueur et l'ordinateur (dans les tableaux X et Y)
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
N, **N1** et **N2**	drapeaux pour déterminer la meilleure direction à prendre par l'ordinateur
T1 et **T2**	total des points du joueur et de l'ordinateur
X1, **X2**, **Y1** et **Y2**	coordonnées des motos du joueur et de l'ordinateur
X() et **Y()**	nombre de caractères de déplacement dans les 4 directions

PROGRAMME

Lignes 30 à 80 : initialisation

Après le dessin du jeu et la définition des caractères graphiques par le sous-programme en 700, on place dans les tableaux X() et Y() le nombre de caractères de déplacement pour les quatre directions.

Lignes 200 à 290 : le jeu

Le caractère saisi au clavier étant stocké dans la variable A$, on modifie suivant sa valeur la variable D1. On utilise pour cela une fonction modulo (MOD) qui évite le test IF THEN beaucoup trop long. D1=(D1+3) MOD 4 remplace D1=D1−1 : IF D1=−1 THEN D1=3 et D1=(D1+1) MOD 4 remplace D1=D1+1 : IF D1=4 THEN D1=0.

Si la moto du joueur rencontre celle d'Amstrad ou le bord de l'écran, le programme va en 610 pour la fin de la partie.

Dans le même cas, pour la moto de l'ordinateur, on va en 400 pour trouver une solution.

Lignes 400 à 520 : mouvements de l'ordinateur

La moto d'Amstrad ayant rencontré un obstacle, le programme cherche une nouvelle direction. S'il n'y a plus de solution, le programme saute à la ligne 600 pour la fin de partie.



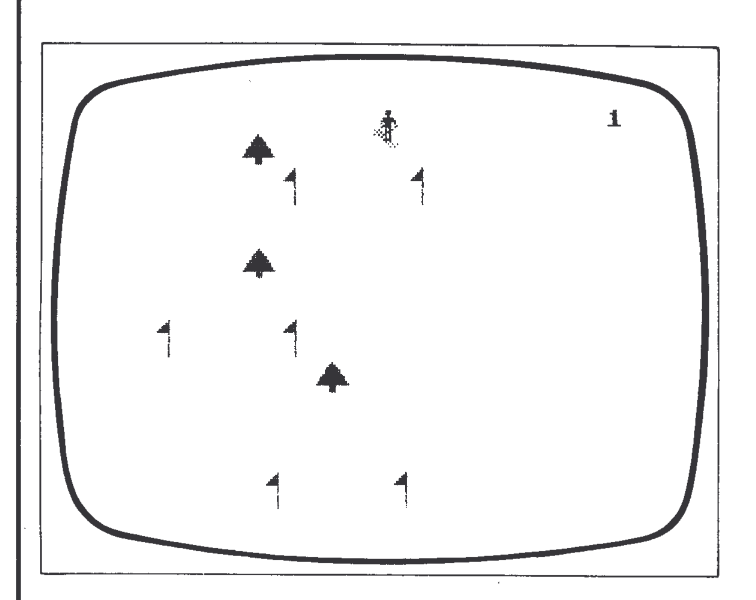
SLALOM

Quoi de plus propice qu'une bonne piste de ski pour exercer ses talents d'agilité et d'endurance ?

En lançant le programme, vous vous retrouverez directement parmi les sapins pour le slalom le plus long du monde.

Le skieur se manœuvre à l'aide des flèches droite et gauche. Le nombre de points gagnés s'inscrit selon les conditions suivantes : une porte passée vaut un point et une porte ratée fait perdre cinq points. Le score s'affiche en permanence en haut à gauche de la piste.

Alors, prêts pour une descente ?



```
10 REM SLALOM
20 REM --------------------------------
30 SYMBOL AFTER 128:GOSUB 1000
40 MODE 1:INK 0,13:INK 1,0:INK 2,6:INK 3,18:PAPER
0:PEN 1:BORDER 13
50 X=20:C=242
80 REM --------------------------------
90 REM LE JEU
100 A=INT(RND*15)+12
110 IF TEST(X*16-8,360)<>0 OR TEST((X+1)*16-8,360)
<>0 THEN 400
120 IF T/8<>INT(T/8) THEN 160
130 PEN 2:LOCATE A,25:PRINT CHR$(140)"       "CHR$(
140)
140 PEN 1:SYMBOL 142,A:PRINT CHR$(142);:PEN 2:PRIN
T TAB(A);CHR$(141)"       "CHR$(141)
150 GOTO 200
160 IF T/6<>INT(T/6) THEN 190
170 PEN 3:LOCATE A,25:PRINT CHR$(136)CHR$(137):PRI
NT TAB(A);CHR$(138)CHR$(139)
180 GOTO 200
190 LOCATE 1,25:PRINT:PRINT
200 T=T+1:PEN 2:LOCATE 36,1:PRINT V:PEN 1
210 C$=INKEY$:IF C$<>"" THEN C=ASC(C$)
220 IF C=242 THEN LOCATE X,1:PRINT CHR$(128)CHR$(1
29)" ":LOCATE X,2:PRINT CHR$(130)CHR$(131)" ":IF X
>11 THEN X=X-1
230 IF C=243 THEN LOCATE X-1,1:PRINT " "CHR$(132)C
HR$(133):LOCATE X-1,2:PRINT " "CHR$(134)CHR$(135):
IF X<29 THEN X=X+1
240 P=0:FOR I=14 TO 0 STEP-2:IF TEST(14-I,382)=1 T
HEN P=P+2↑(I/2)
250 NEXT I:IF P=0 THEN 100
260 IF X>=P AND X<P+7 THEN V=V+1:GOTO 100
270 V=V-5:GOTO 100
380 REM ------------------------------
390 REM PERDU
400 PEN 2:LOCATE 36,1:PRINT V
410 FOR I=50 TO 500 STEP 5:SOUND 1,I,2:NEXT
420 IF INKEY$<>CHR$(13) THEN 420
430 RUN
980 REM ------------------------------
990 REM GRAPHIQUES
1000 SYMBOL 128,3,3,1,7,15,11,19,11
1010 SYMBOL 129,128,128,0,192,224,160,192,160
1020 SYMBOL 130,6,3,2,2,3,2,21,10
1030 SYMBOL 131,148,168,212,160,192,128,0,0
1040 SYMBOL 132,1,1,0,3,7,5,3,5
1050 SYMBOL 133,192,192,128,224,240,208,200,208
1060 SYMBOL 134,41,21,43,5,3,1,0,0
1070 SYMBOL 135,96,192,64,64,192,64,168,80
1080 SYMBOL 136,0,0,1,3,7,15,15,31
1090 SYMBOL 137,0,0,0,128,192,224,224,240
1100 SYMBOL 138,63,63,127,255,3,3,3,0
1110 SYMBOL 139,248,248,252,254,128,128,128,0
1120 SYMBOL 140,1,3,7,15,31,63,1,1
1130 SYMBOL 141,1,1,1,1,1,1,1,1
1140 RETURN
```

PROGRAMME

Lignes 30 à 50 : initialisation

Après la définition des caractères graphiques par le sous-programme en 1000, la ligne 50 initialise à 20 la position de départ du skieur sur l'écran (variable X).

Lignes 100 à 270 : le jeu

Le nombre aléatoire contenu dans A donne la position sur l'écran des prochains piquets ou sapins (ligne 100).

Avant d'afficher sapins ou piquets, on teste par la fonction TEST si l'écran est vide devant le skieur. Si ce n'est pas le cas, on va en 400 pour la fin du jeu.

Le compteur T indique au programme le nombre de lignes déjà passées. Si celui-ci est divisible par 8, on affiche à l'écran deux piquets à partir de la position donnée par A (lignes 120 à 150). Dans la première colonne de cette ligne, on affiche un caractère de code A pour connaître par la suite la position des piquets.

Par contre, si le compteur T est divisible par 6, on effectue la même opération mais cette fois-ci pour un sapin (lignes 160 à 180).

Suivant le code du caractère saisi dans la variable C$, on affiche à la position X un skieur vers la droite ou vers la gauche et l'on incrémente ou l'on décrémente X de la même façon.

La ligne 240 teste le premier caractère de la ligne pour détecter la présence des piquets. Si c'est le cas on compare ce caractère avec la position du skieur. En dehors des limites des piquets, on décrémente de 5 le compteur de points V, sinon on l'incrémente de 1.

LISTE DES VARIABLES :

A	nombre aléatoire : position des piquets ou sapins
C	sens du déplacement 242 = gauche, 243 = droite
C$	dernière touche enfoncée
I	indice de boucle FOR/NEXT
P	caractère sur le bord de l'écran. Indique la présence des piquets
T	nombre de lignes déjà passées
V	nombre de points
X	position du skieur

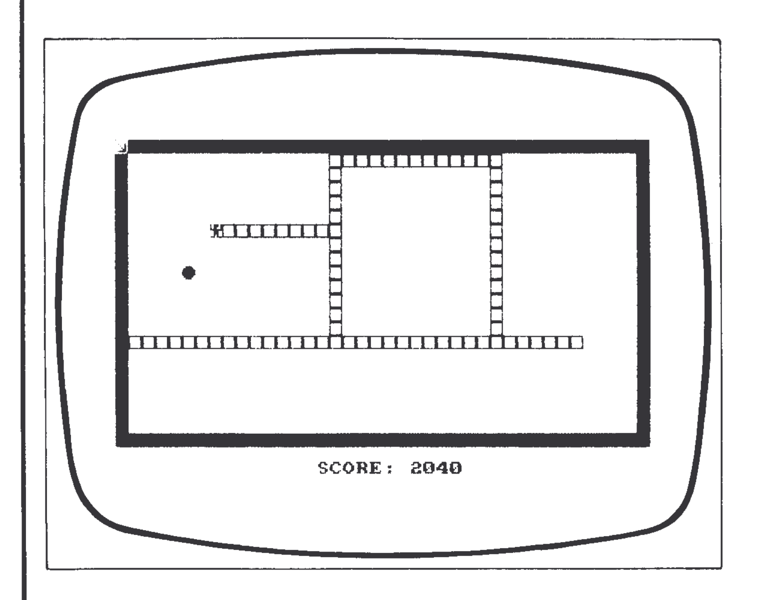


LE BLOB

Il s'agit de bloquer dans le coin en haut à gauche un Blob qui rebondit sur les bords de l'écran.

Pour l'amener à cet endroit précis, vous devez construire un mur en vue de réduire le plus possible son champ de rebond. Les briques se dessinent avec les quatre flèches de déplacement et s'effacent en appuyant simultanément sur SHIFT et l'une des quatre flèches.

Vous avez 5000 fractions de seconde pour neutraliser le Blob et le temps restant constituera votre score.



```
10 REM BLOB
20 REM ------------------------------
30 SYMBOL AFTER 128:MODE 1:INK 0,1:INK 1,24:INK 2,
6:INK 3,15:PAPER 1:PEN 1:BORDER 1
40 SYMBOL 128,128,64,34,18,10,6,62,1
50 SYMBOL 129,60,126,255,255,255,255,126,60
60 SYMBOL 130,255,129,129,129,129,129,129,255
70 SYMBOL 131,93,93,89,127,28,28,20,119
80 FOR I=1 TO 40
90 LOCATE I,1:PRINT" ":LOCATE I,22:PRINT" "
100 IF I<22 THEN LOCATE 1,I:PRINT" ":LOCATE 40,I:P
RINT" "
110 NEXT I:PAPER 0
120 RANDOMIZE TIME
130 LOCATE 1,1:PRINT CHR$(128)
140 X=INT(RND*30)+6:Y=INT(RND*17)+5
150 C=160:T=5000:V=6:W=6:H=1:D=1:F=0
160 LOCATE 16,24:PRINT "SCORE:"
170 GOTO 420
280 REM ------------------------------
290 REM DEPLACEMENT JOUEUR
300 IF TEST(46,352)>0 AND V=2 AND W=2 THEN 800
310 T=T-10:LOCATE 22,24:PRINT T;" ":IF T=0 THEN 70
0
320 A$=INKEY$:IF A$="" THEN 500
330 AX=0:AY=0
340 IF A$=CHR$(242) THEN C=1:AX=-1
350 IF A$=CHR$(243) THEN C=1:AX=1
360 IF A$=CHR$(241) THEN C=1:AY=1
370 IF A$=CHR$(240) THEN C=1:AY=-1
380 IF A$=CHR$(246) THEN C=2:AX=-1
390 IF A$=CHR$(247) THEN C=2:AX=1
400 IF A$=CHR$(244) THEN C=2:AY=-1
410 IF A$=CHR$(245) THEN C=2:AY=1
420 IF TEST((X+AX)*16-8,400-(Y+AY)*16)=1 THEN 500
430 PEN 2:IF C=1 THEN LOCATE X,Y:PRINT CHR$(130) E
LSE LOCATE X,Y:PRINT" "
440 PEN 3:X=X+AX:Y=Y+AY:LOCATE X,Y:PRINT CHR$(131)
480 REM ------------------------------
490 REM MOUVEMENT DU BLOB
500 IF TEST((V+H)*16-2,400-(W+D)*16)=0 THEN 540
510 IF TEST((V+H)*16-2,400-W*16)<>0 AND TEST(V*16-
2,400-(W+D)*16)<>0 THEN D=-D:H=-H:GOTO 300
520 IF TEST((V+H)*16-2,400-W*16)=0 THEN D=-D:IF TE
ST((V+H)*16-2,400-(W+D)*16)=0 THEN 540
530 H=-H:GOTO 300
540 LOCATE V,W:PRINT " "
550 V=V+H:W=W+D
560 PEN 1:LOCATE V,W:PRINT CHR$(129)
570 GOTO 300
680 REM ------------------------------
690 REM PERDU
700 FOR I=50 TO 500 STEP 5:SOUND 1,I,2:NEXT
710 IF INKEY$<>CHR$(13) THEN 710
720 RUN
780 REM -----------------------------
790 REM GAGNE
800 FOR I=500 TO 50 STEP-5:SOUND 1,I,2:NEXT
810 GOTO 710
```

PROGRAMME

Lignes 30 à 170 : initialisation

Après la définition des caractères graphiques, on initialise les diverses variables (voir liste des variables).

Lignes 300 à 440 : déplacement du joueur

Si le Blob est coincé en haut à gauche (V=1 et W=1), le programme va à la ligne 800 pour la phase « gagné ».

A chaque passage dans cette partie du programme, on décrémente le compteur T de 10 points. Si celui-ci est à zéro, on va en ligne 700 pour la fin de partie.

La touche appuyée étant stockée dans la variable A$, on modifie les variables AX, AY et C en fonction de celle-ci.

Lignes 500 à 570 : mouvement du Blob

On déplace continuellement le Blob à 45° sur l'écran tant que l'on ne rencontre pas d'obstacle (bord de l'écran ou mur construit par le joueur). Sinon on modifie les incréments D et H pour prendre une nouvelle direction.

LISTE DES VARIABLES :

A$	dernière touche appuyée
AX et **AY**	déplacement en X et en Y du joueur
C	type de caractère à afficher : C=1 pour les briques, C=2 pour effacer les briques
D et **H**	direction prise par le Blob
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
T	nombre de points
V et **W**	coordonnées du blob
X et **Y**	coordonnées du joueur



PETROLE, PETROLE !

Dans le cadre des économies d'énergie, le conducteur d'une voiture doit faire le plus de kilomètres possible avec une quantité limitée d'essence.

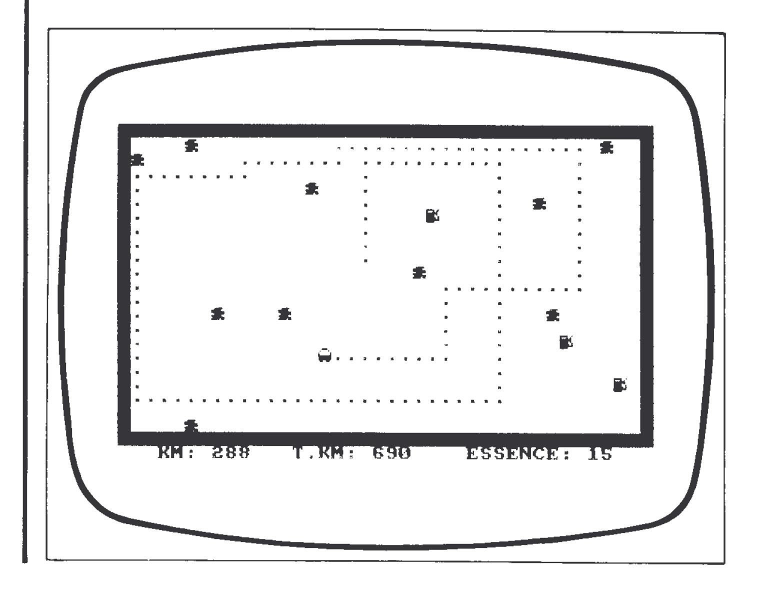
La voiture est dirigée avec les touches P et Q. Elle doit parcourir 400 kilomètres pour passer au tableau suivant.

Mais si, au premier tableau, chaque déplacement vaut trois kilomètres, il n'en vaut plus que deux au deuxième et un au troisième et suivants. **Attention ! Repasser aux mêmes endroits vous fait perdre autant de kilomètres.**

Le nombre de kilomètres par tableau et le total général s'affichent en bas de l'écran ainsi que le contenu du réservoir, sachant que chaque passage à la pompe permet un réapprovisionnement de 10 litres.

Les taches d'huile sur la route ajoutent une difficulté au jeu : en effet, le glissement sur l'une d'elles envoie la voiture à n'importe quel endroit de l'écran.

Il faut garder tout son sang-froid pour la repérer avant qu'elle ne touche le bord du tableau. Dans ce cas, le jeu repart à zéro.



```
10 REM PETROLE, PETROLE !
20 REM ------------------------------
30 SYMBOL AFTER 128:GOSUB 1000
40 N=3:GOTO 100
50 N=N-1:IF N=0 THEN N=1
60 FOR I=500 TO 50 STEP-5:SOUND 1,I,2:NEXT
70 IF INKEY$="" THEN 70
80 REM ------------------------------
90 REM DESSIN DU JEU
100 E=10:K=0
110 DX=0:DY=-1:X=19:Y=10
120 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
 1:PEN 1:BORDER 1
130 FOR I=1 TO 40
140 LOCATE I,1:PRINT" ":LOCATE I,23:PRINT" "
150 IF I<23 THEN LOCATE 1,I:PRINT" ":LOCATE 40,I:P
RINT" "
160 NEXT I:PAPER 0
170 LOCATE X,Y:PRINT CHR$(129)
180 FOR I=1 TO 15
190 W=INT(RND*21)+2:V=INT(RND*38)+2
200 IF TEST(V*16-8,408-W*16)>0 THEN 190
210 IF I<11 THEN PEN 2:LOCATE V,W:PRINT CHR$(130)
ELSE PEN 3:LOCATE V,W:PRINT CHR$(128)
220 NEXT I
230 PEN 1:LOCATE 2,24:PRINT"  KM: 0      T.KM:";KK;
TAB(27)"ESSENCE: 10"
240 IF INKEY$="" THEN 240
280 REM ----------------------------
290 REM LE JEU
300 D$=INKEY$:IF D$="" THEN 380
310 IF DY=0 THEN 350
320 IF D$="Q" THEN DX=DY:GOTO 340
330 DX=-DY
340 DY=0:GOTO 380
350 IF D$="P" THEN DY=DX:GOTO 370
360 DY=-DX
370 DX=0
380 PEN 1:LOCATE X,Y:PRINT "."
400 X=X+DX:Y=Y+DY
410 A=TEST(X*16-8,408-Y*16)
420 IF A=1 THEN 800
430 IF A=3 THEN E=E+10
440 IF TEST(X*16-8,402-Y*16)=1 THEN K=K-2*N:KK=KK-
2*N:IF K<0 THEN 800
450 IF A<>2 THEN 480
460 Y=INT(RND*21)+2:X=INT(RND*38)+2
470 IF TEST(X*16-8,408-Y*16)>0 THEN 460
480 K=K+N:KK=KK+N:E=E-0.1:IF E<=0 THEN 800
490 IF K>=400 THEN 50
```

```
500 LOCATE 7,24:PRINT K;" ":LOCATE 19,24:PRINT KK;
" ":LOCATE 35,24:PRINT INT(E);" "
510 LOCATE X,Y:PRINT CHR$(129)
520 GOTO 300
780 REM ------------------------------
790 REM FIN
800 LOCATE 7,24:PRINT K;" ":LOCATE 19,24:PRINT KK;
" ":LOCATE 35,24:PRINT INT(E+0.001);" "
810 FOR I=50 TO 500 STEP 5:SOUND 1,I,2:NEXT
820 IF INKEY$<>CHR$(13) THEN 820
830 RUN
980 REM ------------------------------
990 REM GRAPHIQUES
1000 SYMBOL 128,249,138,254,250,249,253,255,248
1010 SYMBOL 129,60,66,129,129,255,255,255,102
1020 SYMBOL 130,0,62,252,255,63,252,127,251
1030 RANDOMIZE TIME
1040 RETURN
```

PROGRAMME

Lignes 30 à 70 : initialisation

Après la définition des caractères graphiques par le sous-programme en1000, on place dans la variable N le niveau du jeu.

Lignes 100 à 240 : dessin du jeu

Après l'effacement de l'écran et le dessin du cadre (lignes 120 à 160), on affiche aléatoirement sur l'écran les deux sortes d'éléments (caractères graphiques 128 et 130).

Lignes 300 à 520 : le jeu

Si une touche est appuyée, on modifie en conséquence les incréments de direction DX et DY (lignes 300 à 370). Puis l'on affiche un point à la place de la voiture.

Ensuite, on modifie le contenu de X et Y avec les incréments DX et DY (ligne 400).

On teste si un élément a été rencontré avec la fonction TEST (ligne 410). Si celui-ci est le bord de l'écran (A=1), on va directement en ligne 800 pour la fin du jeu.

Par contre, s'il s'agit d'une pompe (A=3), on incrémente le compteur E de 10.

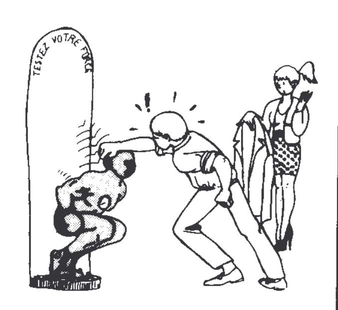
S'il s'agit d'un point, on décrémente le nombre de kilomètres K de 2 fois le niveau.

En ce qui concerne l'huile (A=2), on modifie aléatoirement les coordonnées X et Y de la voiture.

Après l'incrément du compteur K, on teste si celui-ci n'est pas supérieur ou égal à 400. Si c'est le cas, on va en ligne 50 pour changer de niveau, sinon, on affiche la voiture à X,Y, et le jeu continue normalement.

LISTE DES VARIABLES :

A	couleur du point rencontré par la voiture
D$	dernière touche enfoncée
DX et **DY**	sens du déplacement en X et en Y de la voiture
E	nombre de litres d'essence dans la voiture
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
K	nombre de kilomètres parcourus
KK	total des kilomètres de toutes les parties
N	niveau du jeu
V et **W**	coordonnées aléatoires des divers éléments
X et **Y**	coordonnées de la voiture



REFLEXES

Avez-vous de bons réflexes ?

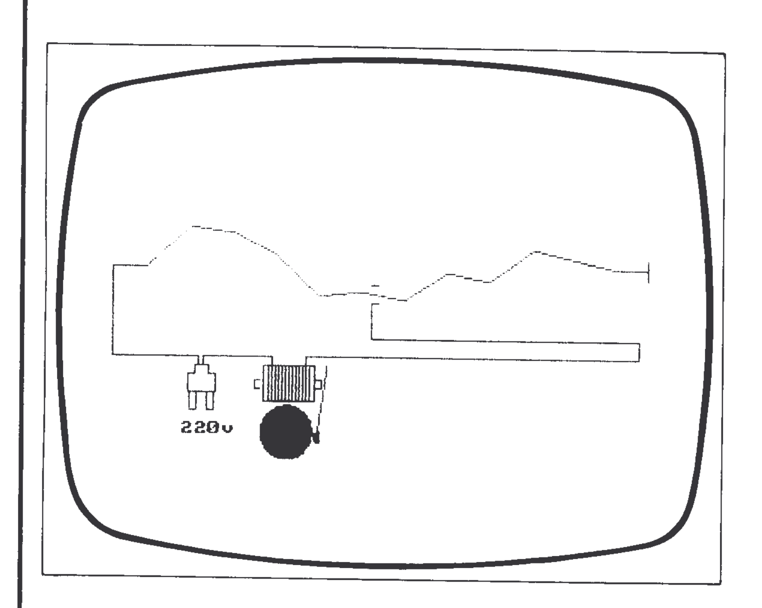
Le jeu suivant vous aidera à le savoir. Il s'agit de guider un anneau le long d'un fil électrique qui dessine des courbes aléatoires plus ou moins compliquées.

Pour cela, vous disposez des flèches haut et bas.

Faites partir l'anneau en tapant sur une touche et rectifiez sans arrêt son glissement alors qu'il a tendance à revenir à l'horizontale.

Une sonnerie intempestive se déclenche quand vous heurtez le fil et quand vous gagnez.

Ce programme est à utiliser de préférence les jours de fête pour tester la lucidité de tous les joyeux convives...



```
10 REM REFLEXES
20 REM ----------------------------
30 MODE 1:INK 0,13:INK 1,0:PAPER 0:PEN 1:BORDER 13
:SPEED KEY 5,3
40 GOSUB 900
50 RANDOMIZE TIME
60 PLOT 0,220:DRAW 40,220
70 X=40:Y=220
80 S=1
90 X=X+50
100 IF RND(1)>0.5 THEN S=-S
110 IY=RND*50*S
120 IF Y+IY<180 OR Y+IY>380 THEN 100
130 DRAW X,Y+IY
140 Y=Y+IY
150 IF X>448 THEN 170
160 GOTO 90
170 DRAW X+100,220
180 DRAW X+134,220
190 PLOT X+134,230:DRAW X+134,208
280 REM ----------------------------
290 REM INITIALISATION CURSEUR
300 X=10:Y=230:P=5:D=1
310 FOR I=1 TO 5
320 X(I)=X:Y(I)=Y
330 PLOT X(I),Y(I)
340 PLOT X(I),Y(I)-20
350 X=X+2
360 NEXT I
370 IF INKEY$="" THEN 370
480 REM ----------------------------
490 REM DEPLACEMENT
500 A$=INKEY$
510 S=2*(A$=CHR$(241))-2*(A$=CHR$(240))
520 IF TEST(X(P)+2,Y(P))=1 THEN 700
530 IF TEST(X(P)+2,Y(P)-18)=1 THEN 700
540 PLOT X(D),Y(D),0:PLOT X(D),Y(D)-20,0
550 PLOT X(D),140:DRAW X(D),Y(D)-20,0
560 X(D)=X(P)+2:Y(D)=Y(P)+S
570 PLOT X(D),Y(D),1:PLOT X(D),Y(D)-20
580 P=P+1:IF P=6 THEN P=1
590 D=D+1:IF D=6 THEN D=1
600 PLOT X(D),140:DRAW X(D),Y(D)-20
610 GOTO 500
680 REM ----------------------------
690 REM SONNETTE
700 IF X(P)>590 THEN 760
710 SOUND 1,600,20
720 FOR I=1 TO 25
730 SOUND 1,60,10:SOUND 1,100,5:NEXT
740 IF INKEY$<>CHR$(13) THEN 740
```

```
750 RUN
760 SOUND 1,600,20
770 FOR I=1 TO 25
780 SOUND 1,600,10:SOUND 1,300,5:NEXT
790 IF INKEY$<>CHR$(13) THEN 790
800 RUN
880 REM --------------------------------
890 REM DESSIN DU JEU
900 FOR I=1 TO 7
910 READ N,X,Y:X=X*2:Y=400-Y*2
920 PLOT X,Y,1
930 FOR J=1 TO N
940 READ X,Y:X=X*2:Y=400-Y*2:DRAW X,Y
950 NEXT J,I
960 FOR I=0 TO 3.14159 STEP 0.06
970 PLOT 204-SIN(I)*30,36-COS(I)*30:DRAW 204+SIN(I
)*30,36-COS(I)*30
980 PLOT 242-SIN(I)*6,30-COS(I)*6:DRAW 242,30-COS(
I)*6
990 NEXT I
1000 FOR I=182 TO 230 STEP 4
1010 PLOT I,110:DRAW I,70
1020 NEXT I
1030 PLOT 250,110:DRAW 240,30
1040 LOCATE 6,23:PRINT"220v"
1050 RETURN
1080 REM ------------------------------
1090 REM GRAPHIQUES
1100 DATA 13,0,90,0,140,50,140,50,145,53,145,53,14
0,93,140,93,145,113,145
1110 DATA 113,140,307,140,307,130,5,130,5,95
1120 DATA 8,48,145,48,150,44,150,44,160,60,160,60,
150,56,150,56,145,48,145
1130 DATA 3,45,160,45,170,49,170,49,160
1140 DATA 3,55,160,55,170,59,170,59,160
1150 DATA 4,88,145,88,165,118,165,118,145,88,145
1160 DATA 3,86,153,83,153,83,158,86,158
1170 DATA 3,119,153,122,153,122,158,119,158
```

PROGRAMME

Lignes 30 à 190 : initialisation

On dessine sur l'écran le jeu à l'aide des DATA en fin de programme (sous-programme en 900). Les lignes 60 à 190 dessinent la ligne brisée en utilisant des nombres aléatoires pour corser le jeu.

Lignes 300 à 370 : initialisation du curseur

Chaque point du curseur est placé dans les tableaux X et Y pour en connaître à tout moment sa position. La variable P pointe le premier de ces points, et D le dernier.

Lignes 500 à 610 : déplacement

La touche appuyée étant placée dans A$, on détermine par une fonction logique la valeur de S (ligne 510).

Les lignes 520 et 530 testent l'écran à côté du curseur pour savoir si la ligne brisée est touchée.

Après l'effacement du dernier point du curseur (à X(D), Y(D)) et de la ligne de raccord électrique, on définit le dernier point comme étant maintenant le premier, et on modifie en conséquence les tableaux X() et Y().

Il ne reste plus qu'à afficher le nouveau point, à incrémenter les pointeurs P et D, et à afficher la ligne de raccord électrique.

Lignes 700 à 800 : sonnette

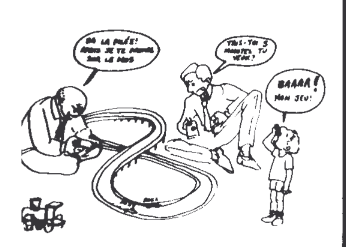
Si l'on a perdu (X(P)<590), on simule une sonnette avec des instructions SOUND.

Lignes 900 à 1050 : dessin du jeu

Les traits qui composent le dessin sont effectués en plusieurs temps. Le premier DATA (variable N) donne le nombre de traits à tracer, les deux suivants le point de départ pour PLOT (variables X et Y), et les N fois 2 DATA qui suivent les coordonnées absolues des instructions DRAW.

LISTE DES VARIABLES :

A$	dernière touche enfoncée
D	pointeur du dernier point du curseur dans les tableaux X() et Y()
I	indice de boucle FOR/NEXT
IY	incrément dans le tracé de la ligne brisée
J	indice de boucle FOR/NEXT
N	nombre de lignes à tracer à partir de PLOT
P	pointeur du premier point du curseur dans les tableaux X() et Y()
S	sens de déplacement : S=−1 pour le haut et S=1 pour le bas
X et **Y**	coordonnées des points de tracé du jeu
X() et **Y()**	coordonnées des points du curseur



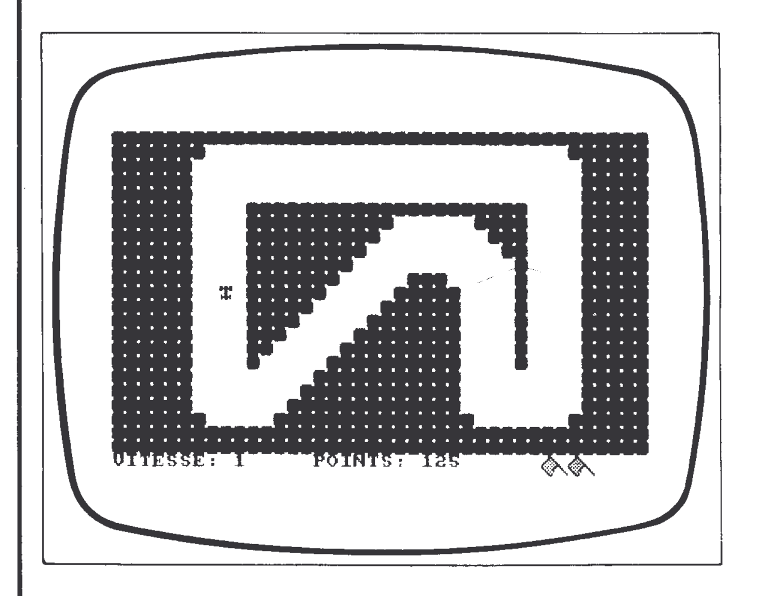
CIRCUIT 24

Trois formes de circuit vous sont proposées aléatoirement.

Vous pouvez les parcourir en déplaçant votre formule 1 à l'aide des touches "/" et "\". Vous disposez aussi de trois vitesses à faire varier avec les touches Z (moins vite) et X (plus vite)

La première vitesse donne 1 point à chaque déplacement, la deuxième, deux points, etc. La vitesse et les scores s'affichent en bas de l'écran.

Trois vies (symbolisées par des drapeaux) vous sont allouées au départ de chaque course. Faites attention, un dérapage est si vite arrivé !



```
10 REM CIRCUIT 24
20 REM ------------------------------
30 SYMBOL AFTER 128:DEFINT A-Z:GOSUB 1000
40 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 3:BORDER 1
50 RANDOMIZE TIME
60 T=2:A=INT(RND*3)+1
70 IF A=1 THEN RESTORE 1100
80 IF A=2 THEN RESTORE 1300
90 IF A=3 THEN RESTORE 1500
100 FOR I=0 TO 22
110 IF I<1 OR I>20 THEN FOR J=1 TO 40:PRINT CHR$(1
30);:NEXT:GOTO 160
120 READ A$:A$="      "+A$
130 FOR J=1 TO 40
140 IF MID$(A$,J,1)="0" THEN PRINT " "; ELSE PRINT
 CHR$(130);
150 NEXT
160 NEXT
170 PEN 2:LOCATE 1,24:PRINT "VITESSE: 1";TAB(16);"
POINTS:";P:PEN 1
180 DX=-1:DY=0:X=26:Y=4:V=1
190 LOCATE X,Y:PRINT CHR$(129)
200 FOR I=1 TO 2
210 LOCATE 31+I*2,24:IF T>=I THEN PRINT CHR$(133)C
HR$(134) ELSE PRINT"  "
220 LOCATE 31+I*2,25:IF T>=I THEN PRINT CHR$(135)C
HR$(136); ELSE PRINT"  ";
230 NEXT I
240 IF INKEY$="" THEN 240
280 REM ------------------------------
290 REM DEPLACEMENTS
300 D$=INKEY$:IF D$="" THEN 400
310 IF D$<>"Z" AND D$<>"X" THEN 350
320 IF D$="Z" AND V>1 THEN V=V-1
330 IF D$="X" AND V<3 THEN V=V+1
340 LOCATE 9,24:PRINT V:GOTO 400
350 IF DY=0 THEN 380
360 IF D$="/" THEN DX=DY ELSE DX=-DY
370 DY=0:GOTO 400
380 IF D$="\" THEN DY=DX ELSE DY=-DX
390 DX=0
400 PEN 1:LOCATE X,Y:PRINT " "
410 X=X+DX:Y=Y+DY
420 IF TEST(X*16-8,408-Y*16)>0 THEN 500
430 LOCATE X,Y:PRINT CHR$(130+DX+DY*2)
440 FOR I=1 TO (3-V)*50:NEXT
450 P=P+V:LOCATE 23,24:PRINT P
460 GOTO 300
480 REM ------------------------------
490 REM FIN
500 FOR I=50 TO 500 STEP 5:SOUND 1,I,2:NEXT:T=T-1:
IF T>-1 THEN 170
510 LOCATE 31,24:PRINT "GAME OVER"
520 IF INKEY$<>CHR$(13) THEN 520
530 RUN
980 REM ------------------------------
990 REM GRAPHIQUES
1000 SYMBOL 128,90,126,90,24,24,219,255,219
1010 SYMBOL 129,231,66,255,255,66,231,7
1020 SYMBOL 130,126,255,255,255,255,255,255,126
1030 SYMBOL 131,224,231,66,255,255,66,231,224
1040 SYMBOL 132,219,255,219,24,24,90,126,90
1050 SYMBOL 133,12,14,27,53,106,213,170,213
1060 SYMBOL 134,0,0,0,128,192,96,240,152
1070 SYMBOL 135,107,53,26,14,4,0,0,0
1080 SYMBOL 136,12,6,3,1,0,0,0,0
```

```
1090 RETURN
1100 DATA"   000000000000000000
1110 DATA"  00000000000000000000
1120 DATA" 0000000000000000000000
1130 DATA"000000000000000000000000
1140 DATA"00000               00000
1150 DATA"0000    000000000     0000
1160 DATA"0000   00090000000     0000
1170 DATA"0000  0000000000000     0000
1180 DATA"0000  00000   00000     00000
1190 DATA"0000  0000     0000      0000
1200 DATA"0000  0000    0000       0000
1210 DATA"0000  0000   0000        0000
1220 DATA"0000  0300  0000         0000
1230 DATA"0000  0000 00000        00000
1240 DATA"0000  0000 0000        00000
1250 DATA"0000  0000 00000      00000
1260 DATA"0000000000 000000000000000
1270 DATA"000000000   0000000000000
1280 DATA" 0000000     00000000000
1290 DATA"  00000       000000000
1300 DATA" 000000000000000000000000000
1310 DATA"00000000000000000000000000000
1320 DATA"00000000000000000000000000000
1330 DATA"00000000000000000000000000000
1340 DATA"0000                     0000
1350 DATA"0000           000000    0000
1360 DATA"0000          00000000   0000
1370 DATA"0000         0000000000  0000
1380 DATA"0000        000000000000 0000
1390 DATA"0000       00000   00000 0000
1400 DATA"0000      00000     0000 0000
1410 DATA"0000     00000      0000 0000
1420 DATA"0000    00000       0000 0000
1430 DATA"0000   00000        0000 0000
1440 DATA"0000  00000         0000 0000
1450 DATA"0000 00000          0000 0000
1460 DATA"000000000           000000000
1470 DATA"00000000            000000000
1480 DATA"0000000             000000000
1490 DATA" 00000               0000000
1500 DATA"                000000000000
1510 DATA"               00000000000000
1520 DATA"               00000000000000
1530 DATA"               00000000000000
1540 DATA"               0000      0000
1550 DATA"               0000      0000
1560 DATA"               0000      0000
1570 DATA"        00000000000      0000
1580 DATA"       000000000000      0000
1590 DATA"      0000000000000      0000
1600 DATA"     0000000000000       0000
1610 DATA"    00000                0000
1620 DATA"   00000                 0000
1630 DATA"  00000                  0000
1640 DATA" 00000                   0000
1650 DATA"00000                    0000
1660 DATA"00000000000000000000000000000
1670 DATA"00000000000000000000000000000
1680 DATA"00000000000000000000000000000
1690 DATA" 000000000000000000000000000
```

PROGRAMME

Lignes 30 à 240 : initialisation

Après la définition des caractères graphiques par le sous-programme en 1000, on place dans la variable T le nombre de vies.

La variable A (nombre aléatoire) détermine le type du circuit. En fonction de celle-ci, on effectue un RESTORE des DATA situées en fin de programme.

Les lignes 100 à 160 affichent à l'écran le dessin du circuit choisi par A en lisant les DATA. Chaque espace est remplacé par un carré noir, les 0 étant remplacés par des espaces. Il est donc possible d'ajouter vos propres circuits à la suite des DATA déjà définies.

Les lignes 200 à 230 affichent le nombre de vies restantes (les petits drapeaux en bas de l'écran).

Lignes 300 à 460 : le jeu

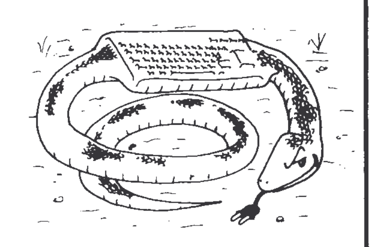
Si une touche est appuyée, on modifie en conséquence la vitesse de la voiture (variable V) ou les incréments de direction DX et DY (lignes 350 à 390). Puis l'on affiche un espace à la place de la voiture.

Ensuite, on modifie le contenu de X et Y avec les incréments DX et DY (ligne 410).

On teste si le bord de la piste a été touché avec la fonction TEST (ligne 420). Si c'est le cas, on va directement à la ligne 500 pour la fin de la partie. Sinon, on affiche la voiture à X,Y et on incrémente le nombre de points P en fonction de la vitesse actuelle.

LISTE DES VARIABLES :

A	type de circuit
A$	chaîne de caractères utilisée pour afficher le jeu
D$	dernière touche enfoncée
DX et **DY**	sens du déplacement en X et en Y de la voiture
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
P	nombre de points
T	nombre de vies
V	vitesse de la voiture (entre 1 et 3)
X et **Y**	coordonnées de la voiture



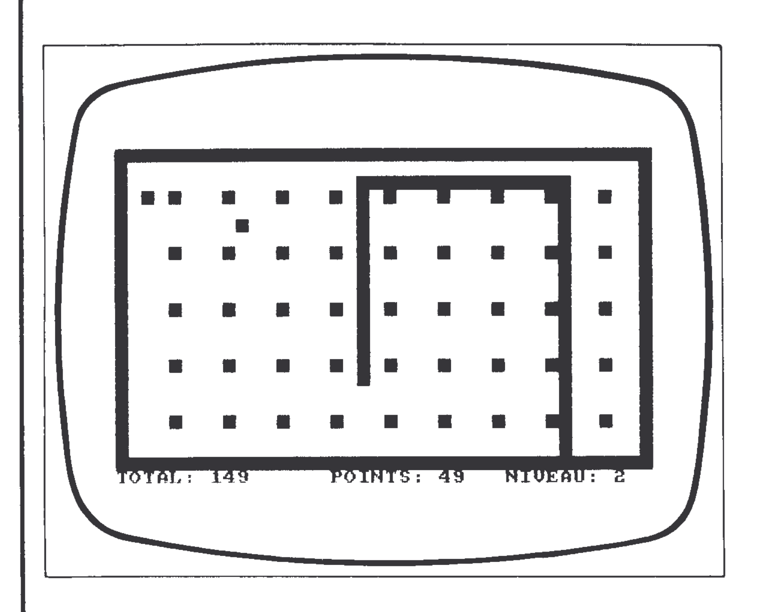
SERPENT INFERNAL

A vous de guider avec dextérité un serpent infernal qui se précipite comme un fou sur tous les obstacles.

Au premier niveau, il faut le manipuler entre quarante-cinq pavés avec les touches P et Q. Chaque déplacement rapporte un point. Vous changez de niveau chaque fois que vous avez atteint cent points (appuyez sur ENTER pour le le tableau suivant). Mais à chaque nouveau niveau un pavé supplémentaire apparaît.

Pour mieux suivre le jeu, Amstrad vous signale en bas de l'écran le nombre de vos points depuis que vous jouez, le nombre de points acquis dans le niveau où vous évoluez et, enfin, ce niveau.

Alors, prêt pour le domptage ?



```
10 REM SERPENT INFERNAL
20 REM ------------------------------
30 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1
40 N=1
50 CLS:RESTORE:GOSUB 600
60 RANDOMIZE TIME
70 X(0)=-1:X(2)=1:Y(1)=1:Y(3)=-1
80 X=19:Y=18:D=3
180 REM -----------------------------
190 REM LE JEU
200 A$=INKEY$
210 IF A$="Q" THEN D=(D+1) MOD 4
220 IF A$="P" THEN D=(D+3) MOD 4
230 X=X+X(D):Y=Y+Y(D)
240 IF TEST(X*16-8,408-Y*16)<>0 THEN 400
250 PAPER 3:LOCATE X,Y:PRINT " ":PAPER 0
260 T=T+1
270 LOCATE 24,24:PRINT T
280 IF T=100 THEN 400
290 GOTO 200
380 REM -----------------------------
390 REM FIN DE PARTIE
400 TT=TT+T
410 LOCATE 7,24:PRINT TT
420 IF INKEY$<>CHR$(13) THEN 420
430 IF T<100 THEN RUN
440 T=0:N=N+1
450 GOTO 50
580 REM -----------------------------
590 REM DESSIN DU JEU
600 PAPER 1
610 FOR I=1 TO 40
620 LOCATE I,1:PRINT" ":LOCATE I,23:PRINT" "
630 IF I<24 THEN LOCATE 1,I:PRINT" ":LOCATE 40,I:P
RINT" "
640 NEXT
650 PAPER 2
660 FOR I=5 TO 37 STEP 4
670 FOR J=4 TO 20 STEP 4
680 LOCATE I,J:PRINT" "
690 NEXT:NEXT
700 FOR I=1 TO N
710 A=INT(RND*38)+2:B=INT(RND*21)+2
720 IF TEST(A*16-8,408-B*16)<>0 THEN 710
730 LOCATE A,B:PRINT" "
740 NEXT
750 PAPER 0
760 LOCATE 1,24:PRINT"TOTAL:";TAB(17);"POINTS:";TA
B(30);"NIVEAU:";N
770 RETURN
```

PROGRAMME

Lignes 30 à 80 : initialisation

On place dans les tableaux X et Y le nombre de caractères de déplacement pour les quatre directions. C'est la ligne 60 qui initialise les nombres aléatoires.

Lignes 200 à 290 : le jeu

La direction du serpent est modifiée en fonction de la dernière touche appuyée. On incrémente alors les coordonnées du serpent X et Y en se référant aux tableaux X( ) et Y( ).

La ligne 240 vérifie si l'on rencontre un objet sur l'écran. Si c'est le cas, on va à la ligne 400 pour la fin du jeu.

Lignes 400 à 450 : fin de partie

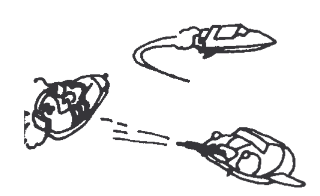
Si le serpent rencontre un objet, on recommence une autre partie (ligne 430), sinon, on incrémente le niveau du jeu et l'on revient en ligne 50 pour un nouveau tableau.

Lignes 600 à 770 : dessin du jeu

Les lignes 610 à 640 affichent le cadre, les lignes 660 à 690 affichent les obstacles et les lignes 700 à 740 affichent les obstacles aléatoires.

LISTE DES VARIABLES :

A et B	coordonnées du point affiché aléatoirement
A$	dernière touche appuyée
D	direction du serpent
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
N	niveau du jeu
T	nombre de points d'un tableau
TT	nombre de points d'une partie
X et **Y**	coordonnées du serpent
X( ) et **Y( )**	nombre de caractères de déplacement dans les quatre directions



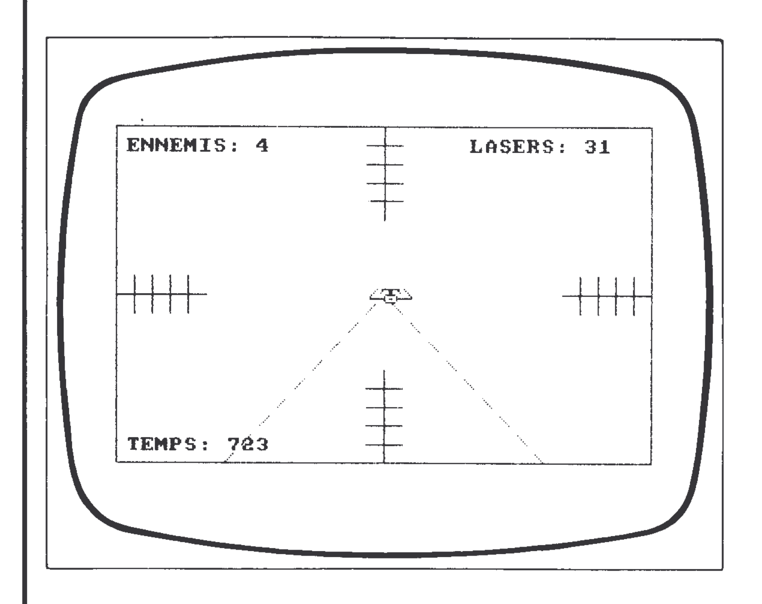
GUERRE DES ETOILES

Les vaisseaux-robots ont envahi l'espace et il devient de plus en plus difficile de voyager vers Mars. Voilà pourquoi les pilotes s'entraînent sur ce simulateur de vol avant de partir en mission.

Dès le départ, Amstrad indique le temps de vol (1000 unités en tout). Le nombre de lasers restants (50 au départ) et d'ennemis abattus apparaît à chaque essai.

Il est possible de viser les vaisseaux en déplaçant votre navette à l'aide des quatre flèches de direction.

Le briefing final se fait dans deux cas: vous n'avez plus de lasers, ou le temps imparti est écoulé. En appuyant sur la touche ENTER vous repartirez pour une autre mission.



```
10 REM GUERRE DES ETOILES
20 REM ------------------------------
30 SYMBOL AFTER 128:GOSUB 1000
40 RANDOMIZE TIME
50 MODE 1:INK 0,3:INK 1,0:PAPER 0:PEN 1:BORDER 3
60 PLOT 20,380,1:DRAW 620,380:DRAW 620,20:DRAW 20,
20:DRAW 20,380
70 PLOT 320,380:DRAW 320,280:PLOT 320,120:DRAW 320
,20
80 PLOT 20,200:DRAW 120,200:PLOT 520,200:DRAW 620,
200
90 FOR I=40 TO 100 STEP 20
100 PLOT 300,400-I:DRAW 340,400-I:PLOT 300,140-I:D
RAW 340,140-I
110 PLOT I,220:DRAW I,180:PLOT I+500,220:DRAW I+50
0,180
120 NEXT I
130 LOCATE 3,23:PRINT"TEMPS:"
140 LOCATE 3,3:PRINT"ENNEMIS:"
150 LOCATE 27,3:PRINT"LASERS:"
180 REM ------------------------------
190 REM LE JEU
200 L=50:T=1000:E=0
210 Z=INT(RND*3)+1
220 X=INT(RND*24)+9:Y=INT(RND*8)+9
230 X=X-1+INT(RND*3):Y=Y-1+INT(RND*3)
240 IF T=0 OR L=0 THEN 700
250 IF X<9 THEN X=9
260 IF X>30 THEN X=30
270 IF Y<9 THEN Y=9
280 IF Y>17 THEN Y=17
290 LOCATE X,Y:PRINT A$(Z)
300 A$=INKEY$:IF A$<>"" THEN 400
310 LOCATE X,Y:PRINT"   "
320 T=T-1:LOCATE 9,23:PRINT T
330 GOTO 230
380 REM ------------------------------
390 REM DEPLACEMENT
400 IF A$=" " THEN 480
410 LOCATE X,Y:PRINT"   "
420 IF A$=CHR$(242) THEN X=X-1
430 IF A$=CHR$(243) THEN X=X+1
440 IF A$=CHR$(240) THEN Y=Y-1
450 IF A$=CHR$(241) THEN Y=Y+1
460 T=T-1:LOCATE 9,23:PRINT T
470 GOTO 240
480 A=TEST(312,206)
490 FOR I=0 TO 160 STEP 20
500 PLOT 140+I,20+I,1:DRAW 150+I,30+I:PLOT 500-I,2
0+I:DRAW 490-I,30+I
510 PRINT CHR$(7);
```

```
520 NEXT I
530 FOR I=0 TO 160 STEP 20
540 PLOT 140+I,20+I,0:DRAW 150+I,30+I:PLOT 500-I,2
0+I:DRAW 490-I,30+I
550 NEXT I
560 L=L-1:LOCATE 34,3:PRINT L
570 IF A=0 THEN 310
580 E=E+1:LOCATE 11,3:PRINT E
590 LOCATE X,Y:PRINT"***":FOR I=500 TO 50 STEP-10:
SOUND 1,I,1:NEXT
600 LOCATE X,Y:PRINT"   "
610 GOTO 210
680 REM ------------------------------
690 REM RESULTATS
700 CLS:PRINT TAB(7);"RESULTATS"
710 PRINT:PRINT:PRINT:PRINT
720 PRINT E;"ENNEMIS DETRUITS":PRINT
730 PRINT " EN";1000-T;"UNITES TEMPS":PRINT
740 PRINT" AVEC";50-L;"LASERS"
750 PRINT:PRINT:PRINT
760 PRINT" MISSION REUSSIE A";INT((E/(50-L))*100);
"%"
770 IF INKEY$<>CHR$(13) THEN 770 ELSE RUN
980 REM ------------------------------
990 REM GRAPHIQUES
1000 SYMBOL 128,28,49,99,255,255,99,49,28
1010 SYMBOL 129,255,129,24,36,36,24,129,255
1020 SYMBOL 130,56,140,198,255,255,198,140,56
1030 SYMBOL 131,9,17,34,64,128,255,0,0
1040 SYMBOL 132,255,24,24,126,129,153,129,126
1050 SYMBOL 133,144,136,68,2,1,255,0,0
1060 SYMBOL 134,31,48,96,192,192,255,0,0
1070 SYMBOL 135,255,0,60,90,165,165,90,60
1080 SYMBOL 136,248,12,6,3,3,255,0,0
1090 A$(1)=CHR$(128)+CHR$(129)+CHR$(130)
1100 A$(2)=CHR$(131)+CHR$(132)+CHR$(133)
1110 A$(3)=CHR$(134)+CHR$(135)+CHR$(136)
1120 RETURN
```

PROGRAMME

Lignes 30 à 150 : initialisation

L'appel du sous-programme en 1000 initialise les caractères graphiques et place dans le tableau A$ le dessin des ennemis.

Les lignes 50 à 120 affichent sur l'écran le dessin de la mire de tir avec des caractères semi-graphiques.

Lignes 200 à 330 : le jeu

La position de départ du vaisseau ennemi est donnée par les valeurs aléatoires contenues dans les variables X et Y. Pour chaque mouvement, on ajoute à celles-ci une valeur comprise entre −1 et 1.

Si le temps imparti est écoulé (T=0) ou s'il ne reste plus de laser (L=0), on va directement à la ligne 700 pour la fin du jeu.

Si aucune touche n'est enfoncée (ligne 300) on revient à la ligne 230 pour modifier à nouveau la position du vaisseau ennemi.

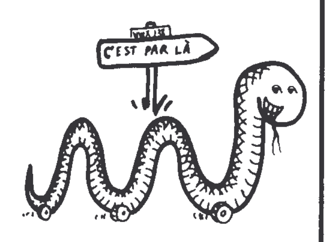
Lignes 400 à 610 : déplacement

Si la touche enfoncée est une flèche, on modifie en fonction de celle-ci les coordonnées X et Y et l'on revient à la ligne 240 pour continuer le jeu.

Dans le cas de la barre d'espace, on affiche le laser. Si l'ennemi est touché, on incrémente le compteur E et l'on revient à la ligne 210 pour changer de type de vaisseau et placer dans X et Y de nouvelles coordonnées aléatoires.

LISTE DES VARIABLES :

A	caractère en dessous du laser
A$	dernière touche enfoncée
A$()	dessin des ennemis
E	nombre d'ennemis abattus
I	indice de boucle FOR/NEXT
L	nombre de lasers
T	temps imparti
Z	type d'ennemi à afficher
X et **Y**	coordonnées du vaisseau ennemi

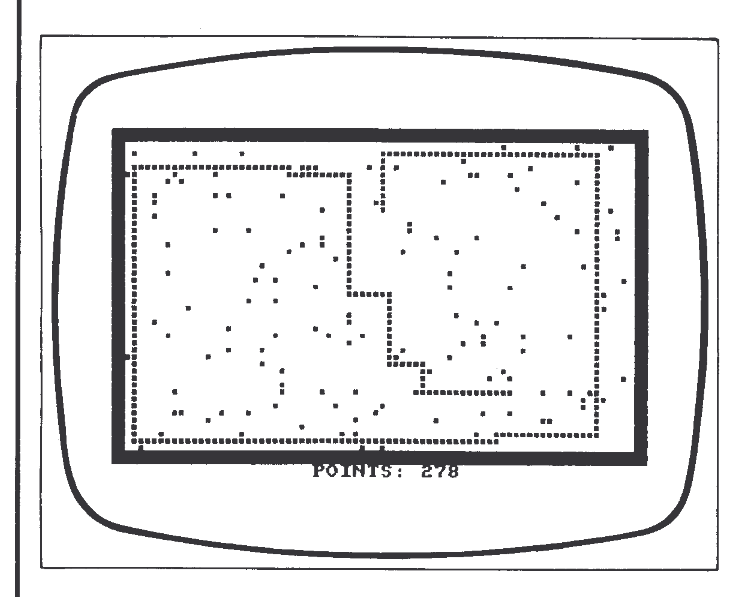


SNAKE

Saurez-vous déplacer un serpent le plus longtemps possible sans qu'il heurte un des innombrables petits carrés qui surgissent devant lui ?

Pour cela, servez-vous des touches P et Q et gardez votre sang-froid.

En cas de collision tapez sur n'importe quelle touche pour une autre partie.



```
10 REM SNAKE
20 REM --------------------------------
30 SYMBOL AFTER 128
40 SYMBOL 128,136,17,34,68,136,17,34,68
50 MODE 1:INK 0,1:INK 1,24:PAPER 1:PEN 1:BORDER 1:
GOSUB 400
60 RANDOMIZE TIME
70 X(0)=-8:X(2)=8:Y(1)=8:Y(3)=-8
80 X=320:Y=296:D=1
180 REM -------------------------------
190 REM LE JEU
200 A$=INKEY$
210 IF A$="P" THEN D=(D+1) MOD 4
220 IF A$="Q" THEN D=(D+3) MOD 4
230 X=X+X(D):Y=Y+Y(D)
240 IF TEST(X,Y)>0 THEN FOR I=50 TO 500 STEP 5:SOU
ND 1,I,2:NEXT:WHILE INKEY$="":WEND:RUN
250 PLOT X+2,Y+2:DRAWR 2,0:DRAWR 0,2:DRAWR -2,0:DR
AWR 0,-2:PLOT X,Y:DRAWR 4,0:DRAWR 0,4:DRAWR -4,0:D
RAWR 0,-4
260 V=INT(RND*152)*8+16:W=400-INT(RND*88)*8
270 IF TEST(V,W)>0 THEN 260
280 PLOT V+2,W+2:DRAWR 2,0:DRAWR 0,2:DRAWR -2,0:DR
AWR 0,-2:PLOT V,W:DRAWR 4,0:DRAWR 0,4:DRAWR -4,0:D
RAWR 0,-4
290 P=P+1:LOCATE 23,25:PRINT P;
300 SOUND 1,40,1
310 GOTO 200
380 REM -------------------------------
390 REM DESSIN DU JEU
400 FOR I=1 TO 40
410 LOCATE I,1:PRINT " ":LOCATE I,24:PRINT " "
420 IF I<25 THEN LOCATE 1,I:PRINT " ":LOCATE 40,I:
PRINT " "
430 NEXT I
440 PAPER 0:LOCATE 16,25:PRINT "POINTS:";
450 RETURN
```

D

Programme

Lignes 30 à 80 : initialisation

Après la définition des caractères graphiques et le dessin du jeu par le sous-programme en 400, on place dans les tableaux X() et Y() le nombre de caractères de déplacement pour les quatre directions.

Lignes 200 à 310 : le jeu

Le caractère saisi au clavier étant stocké dans la variable A$, on modifie suivant sa valeur la variable D. On utilise pour cela une fonction modulo (MOD) qui évite le test IF THEN beaucoup trop long. D=(D+3) MOD 4 remplace D=D−1: IF D=−1 THEN D=3 et D=(D+1) MOD 4 remplace D=D+1: IF D=4 THEN D=0.

Si le serpent rencontre un petit carré ou le bord de l'écran, le programme effectue un RUN dès qu'une touche est appuyée.

Les lignes 260 à 280 affichent à une position aléatoire de l'écran un petit carré. On vérifie avant de l'afficher qu'il n'y a pas d'autre carré ou partie du serpent à cet endroit.

Liste des variables :

A$	dernière touche enfoncée
D	direction prise par le joueur
I	indice de boucle FOR/NEXT
P	total des points
V et **W**	coordonnées du prochain petit carré à afficher
X et **Y**	coordonnées de la tête du serpent
X() et **Y()**	nombre de caractères de déplacement dans les quatre directions.

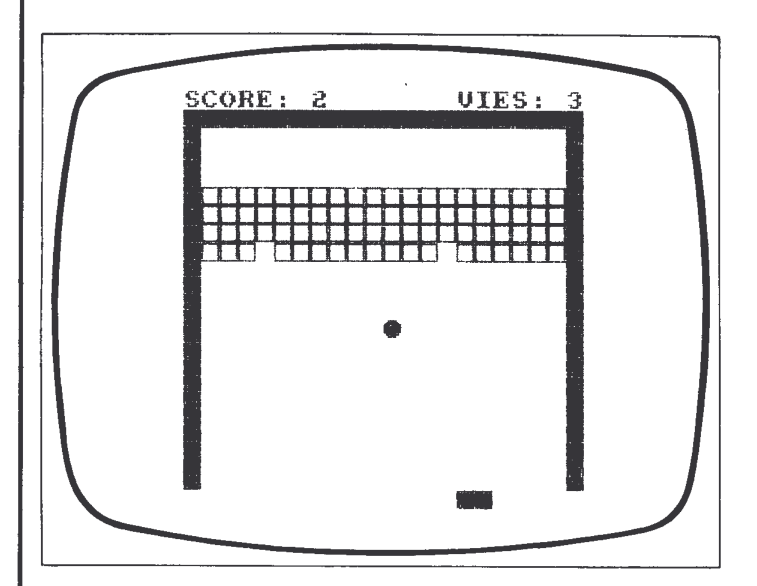


MUR DE BRIQUES

Vous allez pouvoir vous entraîner inlassablement chez vous à ce célèbre jeu vidéo qui consiste à descendre une par une toutes les briques d'un mur.

La balle que vous utilisez pour cela est réceptionnée par la raquette au bas de l'écran. Celle-ci est manœuvrée avec les touches "/" et "\".

Vous n'avez le droit de laisser échapper la balle que trois fois, sinon le mur se redresse et tout est à recommencer.



```
10 REM MUR DE BRIQUES
20 REM ------------------------------
30 DEFINT A-Z:SPEED KEY 5,3
40 RANDOMIZE TIME
50 V=3
60 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1:GOSUB 600
70 CLS
80 LOCATE 25,1:PRINT"VIES:";V
90 FOR I=2 TO 21
100 PAPER 2:LOCATE 9+I,2:PRINT " "
110 LOCATE 10,I:PRINT " "
120 LOCATE 31,I:PRINT " "
130 IF I<10 AND I>5 THEN PAPER 0:PEN 3:LOCATE 11,I
:PRINT STRING$(20,128)
140 NEXT:PAPER 0
150 P=14:T=0
160 X=INT(RND*11)+15:Y=20:DX=-1:DY=-1
280 REM ------------------------------
290 REM LE JEU
300 PEN 1:LOCATE 10,1:PRINT"SCORE:";T+TT
310 A$=INKEY$:P=P+1*(A$="/" AND P>10)-1*(A$="\" AN
D P<28)
320 LOCATE P,22:PRINT" ";:PAPER 1:PRINT"  ";:PAPER
 0:PRINT" "
330 Z=TEST((X+DX)*16-2,408-Y*16):IF Z<>0 THEN SOUN
D 1,200,1:DX=-DX:IF Z=3 THEN SOUND 1,50,1:LOCATE X
-DX,Y:PRINT" ":T=T+1
340 Z=TEST(X*16-2,408-(Y+DY)*16):IF Z<>0 THEN SOUN
D 1,200,1:DY=-DY:IF Z=3 THEN SOUND 1,50,1:LOCATE X
,Y-DY:PRINT" ":T=T+1:GOTO 330 ELSE 330
350 Z=TEST((X+DX)*16-2,408-(Y+DY)*16):IF Z<>0 THEN
 SOUND 1,200,1:DY=-DY:IF Z=3 THEN SOUND 1,50,1:LOC
ATE X+DX,Y-DY:PRINT" ":T=T+1:GOTO 330 ELSE 330
360 A$=INKEY$:P=P+1*(A$="/" AND P>10)-1*(A$="\" AN
D P<28)
370 LOCATE P,22:PRINT" ";:PAPER 1:PRINT"  ";:PAPER
 0:PRINT" "
380 LOCATE X,Y:PRINT" "
390 IF T=80 THEN TT=TT+T:GOTO 70
400 X=X+DX:Y=Y+DY
410 IF Y>21 THEN 500
420 LOCATE X,Y:PRINT CHR$(129)
430 GOTO 300
480 REM ------------------------------
490 REM FIN DE PARTIE
500 V=V-1:LOCATE 25,1:PRINT"VIES:";V
510 IF INKEY$<>CHR$(13) THEN 510
520 IF V<1 THEN RUN
530 LOCATE X,Y:PRINT" "
540 GOTO 160
580 REM ------------------------------
590 REM GRAPHIQUES
600 SYMBOL AFTER 128
610 SYMBOL 128,255,129,129,129,129,129,129,255
620 SYMBOL 129,60,126,255,255,255,255,126,60
630 RETURN
```

PROGRAMME

Lignes 30 à 160 : initialisation

Le DEFINT de la ligne 30 précise au basic que l'on veut des variables entières pour accélérer l'exécution du programme.

Le SPEED KEY de la ligne 30 évite le temps d'attente avant répétition au clavier pour que le déplacement du curseur soit continu.

La boucle des lignes 90 à 140 affiche le jeu. C'est la ligne 130 qui affiche les briques.

Les variables X et Y définissent la position initiale de la balle sur l'écran.

Lignes 300 à 430 : le jeu

C'est la variable P qui détermine la position de la raquette. On la modifie en fonction de la dernière touche enfoncée.

Les lignes 330 à 350 modifient la trajectoire de la balle (incrément DX et DY) en fonction des éléments rencontrés (mur, briques ou raquette).

La ligne 310 est répétée en 360 pour que le curseur se déplace plus rapidement.

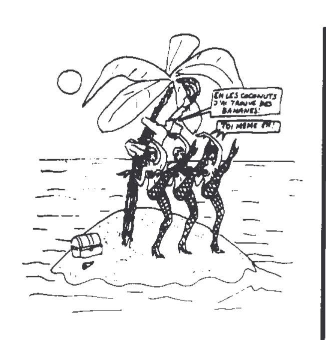
La ligne 410 teste si la balle sort du jeu. Si c'est le cas, le programme va à la ligne 500 pour supprimer une vie.

Lignes 500 à 540 : fin de partie

On décrémente le compteur de vies V. Si celui-ci est à zéro, on reprend une nouvelle partie.

LISTE DES VARIABLES :

A$	dernière touche appuyée
DX et **DY**	direction de la balle
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
P	position du curseur
T	total des points d'un tableau
TT	total des points d'une partie
V	nombre de vies
X et **Y**	position de la balle
Z	caractère sous la balle



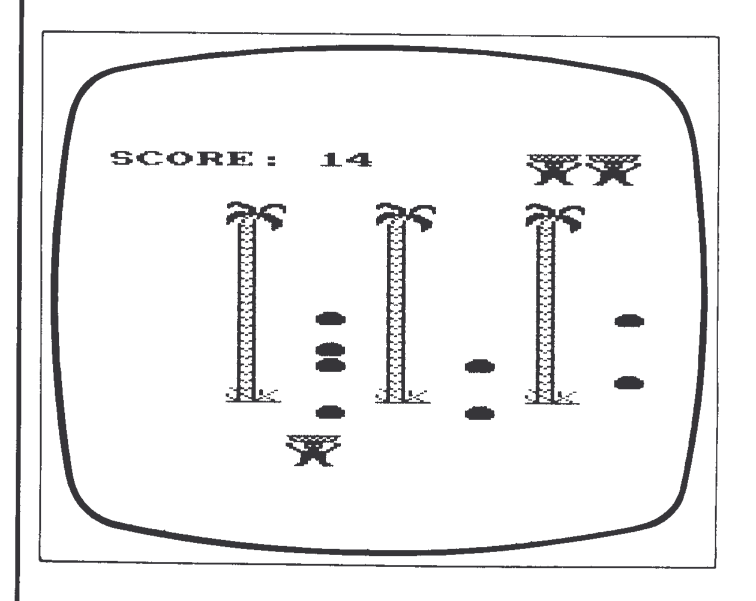
COCONUTS

Essayez d'attraper le plus grand nombre de noix de coco en déplaçant le singe sous les trois cocotiers d'où elles tombent.

Pour cela, vous devez utiliser les touches P et Q.

Votre score s'affiche en haut de l'écran et la fréquence des noix s'accroît toutes les 5 bien réceptionnées.

Trois précieuses vies vous sont attribuées au début d'une partie. Faites-en bon usage. Mais si vous les perdez, le jeu repart tout seul pour un autre entraînement.



```
10 REM COCONUTS
20 REM --------------------------------
30 SPEED KEY 5,3:SYMBOL AFTER 128:GOSUB 3000
40 DIM C(10,3)
50 RANDOMIZE TIME
60 GOSUB 2000
70 P=1:Q=1:D=1:T=0:V=3:N=0.6
80 GOSUB 1000
180 REM ------------------------------
190 REM LE JEU
200 FOR J=1 TO 3
210 A$=INKEY$:P=P+(A$="P")*(P<3)-(A$="Q")*(P>1)
220 PEN 2:LOCATE 7,19:PRINT A$(P):LOCATE 7,20:PRIN
T B$(P)
230 Q=D
240 FOR I=1 TO 9
250 PEN 3:IF C(Q,J)=1 THEN LOCATE J*5+3,I+9:PRINT
CHR$(135) ELSE LOCATE J*5+3,I+9:PRINT " "
260 Q=Q+1:IF Q=11 THEN Q=1
270 NEXT
280 A$=INKEY$:P=P+(A$="P")*(P<3)-(A$="Q")*(P>1)
290 PEN 2:LOCATE 7,19:PRINT A$(P):LOCATE 7,20:PRIN
T B$(P)
300 IF C(Q,J)=0 THEN 360
310 IF P<>J THEN 420
320 PRINT CHR$(7);:T=T+1:IF T/5=INT(T/5) THEN N=N+
0.1
330 C(Q,J)=0
340 A$=INKEY$:P=P+(A$="P")*(P<3)-(A$="Q")*(P>1)
350 PEN 2:LOCATE 7,19:PRINT A$(P):LOCATE 7,20:PRIN
T B$(P)
360 NEXT
370 LOCATE 7,1:PRINT T
380 D=Q
390 IF RND>N THEN 200
400 C(Q,INT(RND*3)+1)=1
410 GOTO 200
420 FOR I=50 TO 500 STEP 10:SOUND 1,I,2:NEXT
430 V=V-1
440 GOSUB 1000
450 IF V>0 THEN 360
460 IF INKEY$<>CHR$(13) THEN 460
470 RUN
980 REM -----------------------------
990 REM AFFICHE LES VIES
1000 PEN 3:FOR K=1 TO 3
1010 IF V<K THEN 1050
1020 LOCATE 13+K*2,1:PRINT LEFT$(A$(1),2)
1030 LOCATE 13+K*2,2:PRINT LEFT$(B$(1),2)
1040 GOTO 1070
1050 LOCATE 13+K*2,1:PRINT "  "
```

```
1060 LOCATE 13+K*2,2:PRINT "  "
1070 NEXT K
1080 RETURN
1980 REM ------------------------------
1990 REM DESSIN DU JEU
2000 MODE 0:INK 0,1:INK 1,24:INK 2,15:INK 3,6:PAPE
R 0:PEN 1:BORDER 1
2010 FOR I=1 TO 3
2020 LOCATE I*5,4:PRINT CHR$(128)CHR$(129):LOCATE
I*5,5:PRINT CHR$(130)CHR$(131)
2030 FOR J=6 TO 15
2040 LOCATE I*5,J:PRINT CHR$(132)
2050 NEXT J
2060 LOCATE I*5,16:PRINT CHR$(133)CHR$(134)
2070 NEXT I
2080 LOCATE 1,1:PRINT "SCORE:"
2090 RETURN
2980 REM ------------------------------
2990 REM GRAPHIQUES
3000 SYMBOL 128,0,112,252,222,143,3,59,127
3010 SYMBOL 129,0,0,30,63,115,97,65,240
3020 SYMBOL 130,250,228,201,201,145,27,21,17
3030 SYMBOL 131,248,188,62,30,14,14,12,8
3040 SYMBOL 132,27,21,17,17,27,21,17,17
3050 SYMBOL 133,91,53,17,153,149,83,49,255
3060 SYMBOL 134,0,68,72,80,32,80,136,254
3070 SYMBOL 135,0,60,126,255,255,255,255,126
3080 SYMBOL 136,255,85,106,63,199,197,103,51
3090 SYMBOL 137,254,84,172,248,198,70,204,152
3100 SYMBOL 138,31,15,15,15,14,12,28,56
3110 SYMBOL 139,240,224,224,224,224,96,112,56
3120 A$(1)=CHR$(136)+CHR$(137)+"          "
3130 B$(1)=CHR$(138)+CHR$(139)+"          "
3140 A$(2)="     "+CHR$(136)+CHR$(137)+"     "
3150 B$(2)="     "+CHR$(138)+CHR$(139)+"     "
3160 A$(3)="          "+CHR$(136)+CHR$(137)
3170 B$(3)="          "+CHR$(138)+CHR$(139)
3180 RETURN
```

PROGRAMME

Lignes 30 à 80 : initialisation

Après la définition des caractères graphiques (sous-programme en 3000), on dimensionne un tableau C pour connaître à tout moment la position des noix de coco sur l'écran.

Le sous-programme en 2000 dessine le jeu et celui en 1000 affiche les vies restantes.

La ligne 70 initialise les diverses variables (voir liste des variables).

Lignes 200 à 470 : le jeu

La boucle J des lignes 200 à 360 permet d'examiner à la suite les trois cocotiers. La boucle I, par contre, permet d'examiner les 9 positions possibles des noix de coco sur chaque arbre.

A chaque boucle J, on teste le clavier et l'on modifie la position du singe (variable P) en utilisant une fonction logique (ligne 210).

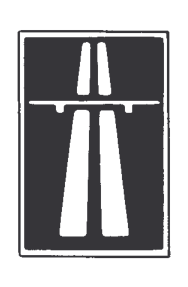
La ligne 310 teste si le singe se trouve bien sous la noix de coco qui arrive sur le sol. Si ce n'est pas le cas, on va directement en ligne 420 pour enlever une vie.

Si le nombre aléatoire qui vient d'être tiré est inférieur au niveau en cours (variable N), on place dans le tableau C une nouvelle noix de coco (lignes 390 à 410).

Les lignes 420 à 470 enlèvent une des vies (variable V). S'il n'en reste plus, le programme effectue un RUN dès qu'une touche est appuyée.

LISTE DES VARIABLES :

A$	dernière touche appuyée
A$() et **B$()**	dessin des singes pour les trois positions (A$=première ligne, B$=deuxième ligne).
C(,)	tableau des positions des noix sur chaque cocotier
D	dernière valeur de Q
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
K	indice de boucle FOR/NEXT
N	niveau du jeu
P	position du singe
Q	pointeur dans le tableau C
T	nombre de points
V	nombre de vies restantes



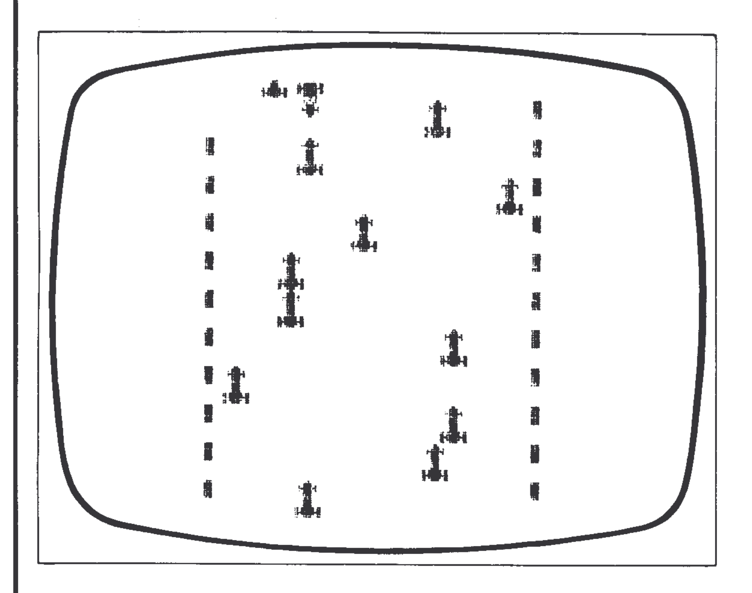
AUTOROUTE

Aidez ce fou du volant qui a pris l'autoroute sur la voie inverse à faire le plus de kilomètres possible sans accident.

Pour lui faire éviter tout ce qui surgit en face de lui, pilotez sa voiture à l'aide des touches P et Q.

Si, par malheur, un accident se produisait quand même, le programme donne alors le nombre de kilomètres parcourus sans accident.

En tapant sur une touche du clavier, vous retrouverez le conducteur imprudent qui a plus que jamais besoin de vous.



```
10 REM AUTOROUTE
20 REM ------------------------------
30 SYMBOL AFTER 128:GOSUB 1000
40 MODE 1:INK 0,1:INK 1,24:INK 2,13:INK 3,6:PAPER
0:PEN 1:BORDER 1
50 RANDOMIZE TIME
60 INPUT "NIVEAU (1 OU 2)";N
70 IF N<1 OR N>2 THEN 60
80 N=N+1
90 X=18
100 P=0
180 REM ------------------------------
190 REM LE JEU
200 A=INT(RND(1)*16)+11
210 PEN 1:LOCATE 10,24:PRINT CHR$(136);:PEN 2:PRIN
T TAB(A);CHR$(132);CHR$(134);
220 PEN 1:PRINT TAB(28);CHR$(136)
230 PEN 2:PRINT TAB(A);CHR$(133);CHR$(135)
240 FOR I=1 TO N:PRINT:NEXT I
250 A$=INKEY$:IF A$<>"" THEN C$=A$
260 IF C$="P" AND X<26 THEN X=X+1
270 IF C$="Q" AND X>11 THEN X=X-1
280 P=P+1
290 PEN 3:LOCATE X,1:PRINT CHR$(128);CHR$(130)
300 PRINT TAB(X);CHR$(129);CHR$(131)
310 IF TEST(X*16,365)=0 AND TEST(X*16-16,365)=0 AN
D TEST(X*16+16,365)=0 THEN 200
380 REM ------------------------------
390 REM FIN
400 FOR I=0 TO 5
410 PRINT CHR$(7)
420 LOCATE X+I+1,I+1:PRINT " ":PRINT TAB(X+I+2);CH
R$(130)
430 LOCATE X-I,I+1:PRINT " ":PRINT TAB(X-I-1);CHR$
(128)
440 LOCATE X+I+1,I+2:PRINT " ":PRINT TAB(X+I+3);CH
R$(131)
450 LOCATE X-I,I+2:PRINT " ":PRINT TAB(X-I-2);CHR$
(129)
460 NEXT I
470 CLS
480 PRINT P;"  KILOMETRES"
490 A$=INKEY$:IF A$="" THEN 490
500 GOTO 90
980 REM ------------------------------
990 REM GRAPHIQUES
1000 SYMBOL 128,2,55,55,63,55,55,7,5
1010 SYMBOL 129,5,6,3,11,15,11,3,1
1020 SYMBOL 130,64,236,236,252,236,236,224,160
1030 SYMBOL 131,160,96,192,208,240,208,192,128
1040 SYMBOL 132,1,3,11,15,11,3,3,3
1050 SYMBOL 133,3,3,3,55,55,63,55,51
1060 SYMBOL 134,128,192,208,240,208,192,192,192
1070 SYMBOL 135,192,192,192,236,236,252,236,204
1080 SYMBOL 136,60,60,60,60,60,60,60,60
1090 RETURN
```

Programme

Lignes 30 à 100 : initialisation

On définit les caractères graphiques des voitures par le sous-programme en 1000.

La variable X donne la position de départ de la voiture du joueur et P le nombre de kilomètres parcourus.

La variable N saisie au clavier détermine la rapidité du jeu.

Lignes 200 à 310 : le jeu

Le nombre aléatoire contenu par la variable A donne l'emplacement de la voiture en sens inverse. Celle-ci est affichée simplement par une fonction PRINT ce qui a pour effet de faire monter l'écran de deux lignes vers le haut.

La ligne 250 teste le sens de déplacement donné par la touche tapée au clavier.

Si un déplacement est effectué, on modifie la valeur de la variable X avant d'afficher à l'écran le dessin de la voiture (lignes 260 à 300).

Si des caractères différents de 0 (couleur du fond) sont en face de la voiture du joueur (testés par TEST), on va directement à la fin du jeu (ligne 400).

Lignes 400 à 500 : fin de partie

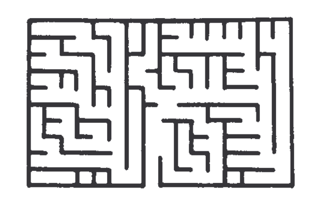
Des caractères différents de 0 se trouvant devant la voiture du joueur, on affiche le nombre de kilomètres parcourus avant de revenir à une autre partie dès qu'une touche est enfoncée.

Liste des variables :

A	position de la voiture en face
A$	dernière touche enfoncée
C$	sens du déplacement
I	indice de boucle FOR/NEXT
N	niveau du jeu
P	nombre de kilomètres parcourus
X	position de la voiture du joueur

CHAPITRE II :

JEUX DE REFLEXION



LABYRINTHE EN TROIS DIMENSIONS

Saurez-vous sortir de ce labyrinthe mystérieux ?

Il comporte en tout 400 pièces (20 en largeur et 20 en longueur) et s'affiche en trois dimensions.

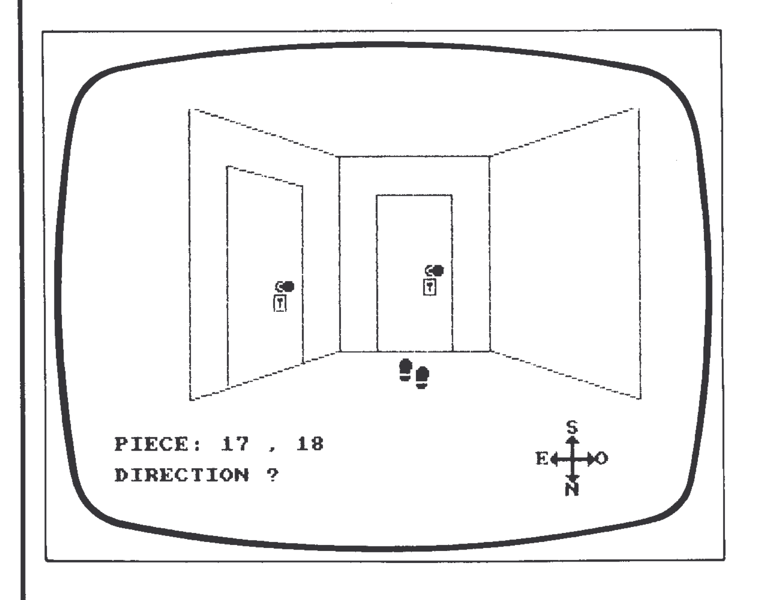
Amstrad vous indique toujours la pièce où vous vous trouvez et dessine en plus une boussole pour ne pas vous perdre !

Si vous faites demi-tour, la trace de vos pas reste imprimée sur le sol.

Il suffit de rentrer l'initiale de la direction choisie pour que l'ordinateur dessine la pièce où vous venez d'arriver.

Dès que vous aurez trouvé la sortie, tapez une dernière fois sa direction et vous obtiendrez la vue d'ensemble du labyrinthe avec toutes les pièces que vous avez visitées.

Ne croyez pas retraverser le labyrinthe plus rapidement que la première fois. Tous les labyrinthes sont différents !



```
10 REM LABYRINTHE 3D
20 REM -------------------------------
30 SYMBOL AFTER 128:GOSUB 1400
40 DEFINT A-Z
50 RANDOMIZE TIME
60 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1:LOCATE 10,11:PRINT "GENERATION DU
 LABYRINTHE"
70 DIM A(21,21),B(20,20)
80 S(0)=4:S(1)=2:S(2)=1:S(3)=8:S$(0)="N":S$(1)="E"
:S$(2)="S":S$(3)="O"
90 SS$(0)="AU NORD  ":SS$(1)="A L'EST  ":SS$(2)="A
U SUD   ":SS$(3)="A L'OUEST"
100 S=2:Z(1)=22:Z(2)=20:Z(3)=22
110 X(1)=1:X(3)=-1:Y(0)=-1:Y(2)=1
120 P$=CHR$(137)+CHR$(138)+CHR$(10)+CHR$(8)+CHR$(8
)+CHR$(139)+CHR$(140)
130 A$=CHR$(10)+CHR$(8):S$="S"+A$+"O"+A$+"R"+A$+"T
"+A$+"I"+A$+"E"
180 REM -------------------------------
190 REM CREATION
200 X=10:Y=1:DE=10
210 A(X,Y)=A(X,Y)+4
220 D=INT(RND*5)+1
230 ON D GOTO 240,270,300,330,300
240 IF Y-1=0 THEN 220
250 IF A(X,Y-1) AND 4 THEN 360
260 Y=Y-1:A(X,Y)=A(X,Y)+1:GOTO 400
270 IF X+1=21 THEN 220
280 IF A(X+1,Y) AND 4 THEN 360
290 A(X,Y)=A(X,Y)+2:X=X+1:GOTO 400
300 IF A(X,Y+1) AND 4 THEN 360
310 A(X,Y)=A(X,Y)+1:Y=Y+1:IF Y=21 THEN 500
320 GOTO 400
330 IF X-1=0 THEN 220
340 IF A(X-1,Y) AND 4 THEN 360
350 X=X-1:A(X,Y)=A(X,Y)+2:GOTO 400
360 F=F+1:IF F<20 THEN 220
370 F=0:F1=F1+1:X=DE:Y=1:IF F1<9 THEN 220
380 F1=0:F2=F2+1:IF F2<19 THEN 200
390 RUN
400 A(X,Y)=A(X,Y)+4:F=0:GOTO 220
480 REM -------------------------------
490 REM DESTRUCTION
500 DA=X
510 FOR J=1 TO 20
520 Z=INT(RND*5)+1
530 FOR I=1 TO 20
540 IF A(I,J)>3 AND Z<3 THEN 640
550 A=INT(RND*2)+1:ON A GOSUB 580,610
```

```
560 IF B=0 AND I<>20 AND J<>20 THEN ON A GOSUB 610
,580
570 B=0:GOTO 640
580 IF J=20 OR A(I,J+1)>3 OR (A(I,J)=1 AND Z>3) TH
EN GOTO 600
590 A(I,J)=A(I,J)+1:B=1
600 RETURN
610 IF I=20 OR A(I+1,J)>3 OR (A(I,J)=2 AND Z>3) TH
EN 630
620 A(I,J)=A(I,J)+2:B=1
630 RETURN
640 NEXT
650 NEXT
660 FOR I=1 TO 20:FOR J=1 TO 20
670 IF A(I,J) AND 4 THEN A(I,J)=A(I,J)-4
680 NEXT:NEXT
690 FOR I=1 TO 20:FOR J=1 TO 20
700 IF A(I-1,J) AND 2 THEN A(I,J)=A(I,J)+8
710 IF A(I,J-1) AND 1 THEN A(I,J)=A(I,J)+4
720 NEXT:NEXT
780 REM ------------------------------
790 REM LE JEU
800 X=10:Y=1
810 CLS
820 PLOT 240,350,1:DRAW 400,350:DRAW 400,150:DRAW
240,150:DRAW 240,350
830 PLOT 240,350:DRAW 80,400:DRAW 80,100:DRAW 240,
150
840 PLOT 400,350:DRAW 560,400:DRAW 560,100:DRAW 40
0,150
850 LOCATE 31,22:PRINT CHR$(132):LOCATE 30,23:PRIN
T CHR$(133)CHR$(134)CHR$(135);:LOCATE 31,24:PRINT
CHR$(136):LOCATE 31,25:PRINT S$((S+2) MOD 4)
860 LOCATE 1,22:PRINT"PIECE:";X","Y
870 IF B(X,Y)=1 THEN LOCATE 20,17:PRINT CHR$(128)C
HR$(129):LOCATE 20,18:PRINT CHR$(130)CHR$(131)
880 B(X,Y)=1
890 FOR J=1 TO 3
900 IF J=1 THEN I=(S+3) MOD 4
910 IF J=2 THEN I=S
920 IF J=3 THEN I=(S+1) MOD 4
930 IF A(X,Y) AND S(I) THEN ON J GOTO 940,970,1000
 ELSE 1020
940 PLOT 120,116:DRAW 120,340:DRAW 200,320:DRAW 20
0,140:LOCATE 11,12:PRINT P$
950 IF X=DA AND Y=20 AND I=2 THEN LOCATE 9,8:PRINT
 S$
960 GOTO 1020
970 PLOT 280,310:DRAW 360,310:DRAW 360,150:DRAW 28
0,150:DRAW 280,310:LOCATE 21,11:PRINT P$
```

```
980 IF X=DA AND Y=20 AND I=2 THEN LOCATE 18,5:PRIN
T"SORTIE"
990 GOTO 1020
1000 PLOT 520,116:DRAW 520,340:DRAW 440,320:DRAW 4
40,140:LOCATE 31,12:PRINT P$
1010 IF X=DA AND Y=20 AND I=2 THEN LOCATE 29,8:PRI
NT LEFT$(S$,8):LOCATE 30,11:PRINT RIGHT$(S$,8)
1020 LOCATE 27+J*2,Z(J)+1:PRINT S$(I)
1030 NEXT J
1040 LOCATE 1,24:PRINT"DIRECTION ?";:A$="":WHILE A
$="":A$=INKEY$:WEND
1050 S=-1
1060 FOR I=0 TO 3
1070 IF A$=S$(I) THEN S=I
1080 NEXT I:IF S<0 THEN 1040
1090 IF A(X,Y) AND S(S) THEN 1110
1100 LOCATE 1,25:PRINT "PAS DE PORTE ";SS$(S);:GOT
O 1040
1110 X=X+X(S):Y=Y+Y(S)
1120 IF Y=21 THEN GOSUB 1200:RUN
1130 GOTO 810
1180 REM ----------------------------
1190 REM DESSIN DU LABYRINTHE
1200 CLS
1210 FOR J=1 TO 20
1220 FOR I=1 TO 20
1230 IF B(I,J)=1 THEN PAPER 2 ELSE PAPER 0
1240 LOCATE I+1,J+1:PRINT " "
1250 PLOT I*16,400-J*16,1:DRAW I*16+14,400-J*16:DR
AW I*16+14,386-J*16:DRAW I*16,386-J*16:DRAW I*16,4
00-J*16
1260 IF B(I,J)=1 THEN C=2 ELSE C=0
1270 IF A(I,J) AND 1 THEN PLOT I*16+2,386-J*16,C:D
RAW I*16+12,386-J*16
1280 IF A(I,J) AND 2 THEN PLOT I*16+14,398-J*16,C:
DRAW I*16+14,388-J*16
1290 IF A(I,J) AND 4 THEN PLOT I*16+2,400-J*16,C:D
RAW I*16+12,400-J*16
1300 IF A(I,J) AND 8 THEN PLOT I*16,398-J*16,C:DRA
W I*16,388-J*16
1310 NEXT:NEXT
1320 PEN 1
1330 IF INKEY$<>CHR$(13) THEN 1330
1340 RETURN
1380 REM ----------------------------
1390 REM GRAPHIQUES
1400 SYMBOL 128,56,124,254,254,254,254,252,252
1410 SYMBOL 129,0,0,0,0,28,62,127,127
1420 SYMBOL 130,0,252,252,120,0,0,0,0
1430 SYMBOL 131,127,127,126,126,0,126,126,60
1440 SYMBOL 132,24,60,126,255,24,24,24,24
```

```
1450 SYMBOL 133,16,48,112,255,255,112,48,16
1460 SYMBOL 134,24,24,24,255,255,24,24,24
1470 SYMBOL 135,8,12,14,255,255,14,12,8
1480 SYMBOL 136,24,24,24,24,255,126,60,24
1490 SYMBOL 137,1,3,2,2,3,1,0,7
1500 SYMBOL 138,222,63,255,255,63,222,0,240
1510 SYMBOL 139,4,5,5,4,4,4,4,7
1520 SYMBOL 140,16,208,208,144,144,144,16,240
1530 RETURN
```

LISTE DES VARIABLES :

A(,)	contient la représentation du labyrinthe
B(,)	B(X,Y)=1 si la pièce a déjà été visitée
A$	choix du joueur
B	B=1 si l'on a déjà détruit une partie de la case
D	sens du déplacement dans la création
DA	garde le numéro de la case d'arrivée
DE	garde le numéro de la case de départ
F,F1 et **F2**	compteurs indiquant le blocage du chemin gagnant
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
S	sens du déplacement du joueur
S()	valeur binaire correspondant aux quatre directions
S$()	choix possible du joueur
SS$()	idem à S$, mais en clair
X et **Y**	coordonnées de la pièce où se trouve le joueur
X() et **Y()**	nombre de cases de déplacement dans les quatre directions
Z	si Z>2 on détruit la case adjacente au chemin gagnant
Z()	coordonnées sur l'écran de la boussole

PROGRAMME

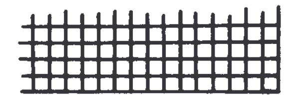
Lignes 30 à 130 : initialisation

Après la définition des caractères graphiques, on initialise divers paramètres (voir liste des variables).

Lignes 200 à 400 : création

On crée dans le tableau A un chemin gagnant en partant de la case 10,1 jusqu'à la fin du tableau. Les cases sont représentées ainsi :

On remarque que l'assemblage des cases donne une grille :



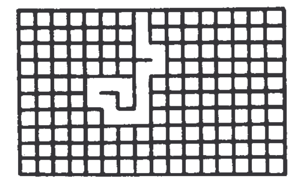
Chaque case peut contenir un de ces quatre chiffres :

binaire	décimal	forme de la case
00	0	la case est complète
01	1	on supprime la barre du bas
10	2	on supprime la barre du haut
11	3	la case est vide

En effet, on utilise le poids binaire de chaque case pour en déterminer la forme. 0 correspond à une case complète, ce qui évite d'initialiser le tableau A au début du jeu. On gagne ainsi un temps précieux

De plus, on ajoute 4 (100 binaire, ce qui ne change pas le résultat du tableau précédent) pour chaque case correspondant au chemin gagnant. On peut alors vérifier si l'on est déjà passé sur une case en testant le troisième bit.

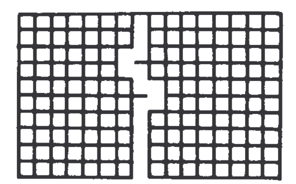
Il arrive parfois qu'un chemin soit bloqué s'il boucle sur lui-même :



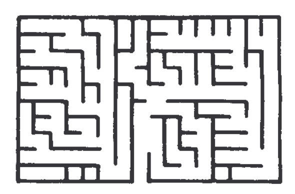
Dans ce cas on revient à la première ligne pour créer un nouveau chemin (rôle des compteurs F, F1 et F2).

Lignes 500 à 720 : destruction

Si l'on éditait tout de suite le labyrinthe, on obtiendrait quelque chose dans ce genre :



Il faut donc détruire toutes les autres cases en supprimant soit la barre verticale, soit la barre horizontale. Dans le cas des cases adjacentes au chemin gagnant, on supprime une de ces barres si la variable aléatoire Z est supérieure à 3, ce qui ajoute des « culs-de-sac ».



labyrinthe de l'exemple précédent après destruction

Lignes 800 à 1130 : le jeu

Le programme dessine à l'écran la pièce X,Y à partir du tableau A. La variable DA donnant le point d'arrivée, la ligne 950 ou 980 ou 1010 affiche le mot SORTIE au-dessus de la porte concernée.

Si vous êtes déjà passé dans une pièce (B(X,Y)=1), on affiche en plus la trace de vos pas.

Les lignes 1040 à 1130 saisissent la direction que vous voulez prendre. En cas d'impossibilité, on revient en ligne 1040.

Lignes 1200 à 1340 : dessin du labyrinthe

Si vous avez trouvé la sortie (Y=21), le programme affiche le labyrinthe complet vu de dessus et en mettant un point dans les pièces où vous êtes passé (renseignements donnés par le tableau B).



CHIEN DE BERGER

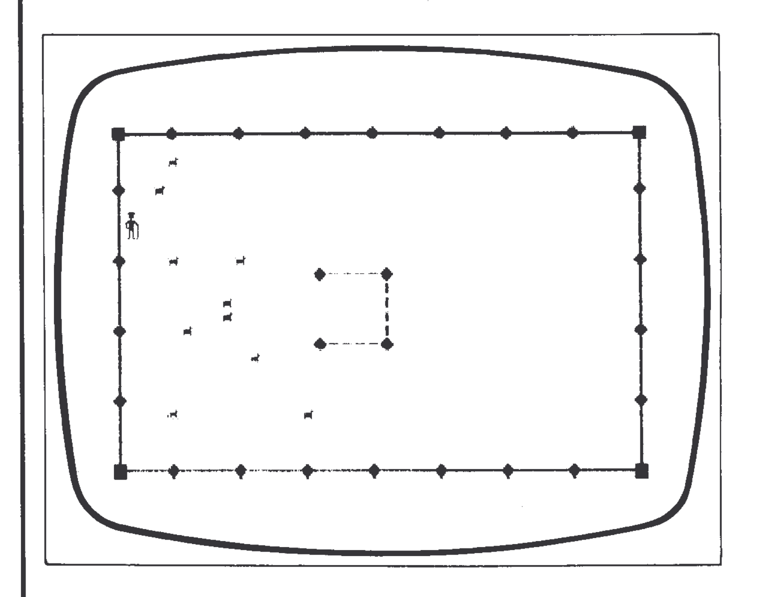
Ce jeu a pour inspiration les fameuses compétitions de chiens de berger en Ecosse.

Le gagnant est celui qui ramène le plus vite possible tout un troupeau de moutons dans son enclos.

Ici le chien de berger peut diriger de cinq à neuf moutons selon les parties. Vous devez le déplacer avec les quatre flèches et valider le mouvement en tapant sur ENTER.

A ce moment tous les moutons bougent et seuls ceux qui sont vraiment près de l'enclos y pénètrent définitivement. Les autres s'éparpillent de plus en plus dans le pâturage et le pauvre chien n'a plus qu'à les rabattre à nouveau vers l'enclos.

Bien sûr, il s'agit d'utiliser le moins de coups possible. Le nombre d'essais s'affiche d'ailleurs dès que tous les moutons sont dans l'enclos.



```
10 REM CHIEN DE BERGER
20 REM --------------------------------
30 MODE 1:INK 0,1:INK 1,15:PAPER 0:PEN 1:BORDER 1
40 DIM X(10),Y(10)
50 F=0
60 E=0
70 SYMBOL AFTER 128
80 SYMBOL 128,0,4,6,124,188,164,18,0
90 SYMBOL 129,0,2,67,62,62,62,34,0
100 SYMBOL 130,24,60,126,255,255,126,60,24
110 SYMBOL 131,56,124,56,56,16,124,254,187
120 SYMBOL 132,185,249,57,41,41,41,41,239
130 SYMBOL 133,255,255,255,255,255,255,255,255
140 SYMBOL 134,0,24,24,24,24,24,24,0
150 RANDOMIZE TIME
180 REM -----------------------------
190 REM DESSIN DU JEU
200 PLOT 6,6:DRAW 632,6:DRAW 632,392:DRAW 6,392:DR
AW 6,6
210 PLOT 8,8:DRAW 630,8:DRAW 630,390:DRAW 8,390:DR
AW 8,8
220 LOCATE 1,1:PRINT CHR$(133):LOCATE 40,1:PRINT C
HR$(133)
230 LOCATE 1,25:PRINT CHR$(133);:LOCATE 40,25:PRIN
T CHR$(133);
240 FOR I=5 TO 38 STEP 5
250 LOCATE I,1:PRINT CHR$(130):LOCATE I,25:PRINT C
HR$(130);
260 IF I<25 THEN LOCATE 1,I:PRINT CHR$(130):LOCATE
 40,I:PRINT CHR$(130)
270 NEXT I
280 REM -----------------------------
290 REM AFFICHE BERGER,CHIEN,MOUTONS
300 X=INT(RND(1)*13)+2
310 Y=INT(RND(1)*19)+2
320 LOCATE X,Y:PRINT CHR$(131):LOCATE X,Y+1:PRINT
CHR$(132)
330 V=INT(RND(1)*10)+16:W=INT(RND(1)*15)+2
340 LOCATE V,W:PRINT CHR$(130);"----";CHR$(130)
350 FOR I=W+1 TO W+4:LOCATE V+5,I:PRINT CHR$(134):
NEXT I
360 LOCATE V,W+5:PRINT CHR$(130);"----";CHR$(130)
370 W=W+1
380 N=5+INT(RND(1)*5)
390 FOR I=1 TO N+1
400 X(I)=INT(RND(1)*14)+2
410 Y(I)=INT(RND(1)*20)+2
420 IF TEST(X(I)*16-8,408-Y(I)*16)<>0 THEN 400
430 IF I=N+1 THEN LOCATE X(I),Y(I):PRINT CHR$(128)
:X=X(I):Y=Y(I):GOTO 450
440 LOCATE X(I),Y(I):PRINT CHR$(129)
450 NEXT I
480 REM -----------------------------
490 REM DEPLACE CHIEN ET MOUTONS
500 D$=INKEY$:IF D$="" THEN 500
510 IF D$=CHR$(13) THEN E=E+1:GOTO 580
520 LOCATE X,Y:PRINT " "
```

```
530 IF D$=CHR$(242) THEN IF TEST((X-1)*16-8,408-Y*
16)=0 THEN X=X-1
540 IF D$=CHR$(243) THEN IF TEST((X+1)*16-8,408-Y*
16)=0 THEN X=X+1
550 IF D$=CHR$(240) THEN IF TEST(X*16-8,408-(Y-1)*
16)=0 THEN Y=Y-1
560 IF D$=CHR$(241) THEN IF TEST(X*16-8,408-(Y+1)*
16)=0 THEN Y=Y+1
570 LOCATE X,Y:PRINT CHR$(128):GOTO 500
580 FOR J=1 TO 5
590 F=0
600 FOR I=1 TO N
610 LOCATE X(I),Y(I):PRINT " "
620 IF X(I)<V OR X(I)>V+4 THEN 680
630 IF Y(I)<W OR Y(I)>W+3 THEN 680
640 F=F+1
650 IF TEST((X(I)+1)*16-8,408-Y(I)*16)<>0 THEN 790
660 X(I)=X(I)+1
670 GOTO 790
680 AX=X(I):AY=Y(I)
690 IF Y(I)=2 THEN Y(I)=3:GOTO 770
700 IF Y(I)=24 THEN Y(I)=23:GOTO 770
710 IF Y<Y(I) THEN Y(I)=Y(I)+1
720 IF Y>Y(I) THEN Y(I)=Y(I)-1
730 IF X(I)=2 THEN X(I)=3:GOTO 770
740 IF X(I)=39 THEN X(I)=38:GOTO 770
750 IF X<X(I) THEN X(I)=X(I)+1
760 IF X>X(I) THEN X(I)=X(I)-1
770 IF TEST(X(I)*16-8,408-Y(I)*16)=0 THEN 790
780 X(I)=AX:Y(I)=AY
790 LOCATE X(I),Y(I):PRINT CHR$(129)
800 NEXT I
810 IF F=N THEN 900
820 NEXT J
830 GOTO 500
880 REM -----------------------------
890 REM FIN
900 LOCATE 15,23:PRINT E;"ESSAIS";
910 FOR I=1 TO 20:PRINT CHR$(7);:FOR J=1 TO 20:NEX
T:NEXT
920 IF INKEY$="" THEN 920
930 RUN
```

Liste des variables :

AX et **AY**	anciennes coordonnées d'un mouton avant déplacement
D$	dernière touche appuyée
E	nombre d'essais
F	nombre de moutons dans la bergerie
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
N	nombre de moutons
V et **W**	coordonnées de la bergerie
X et **Y**	coordonnées du chien
X() et **Y()**	coordonnées des moutons

Programme

Lignes 30 à 150 : initialisation

Après le dimensionnement des tableaux X et Y pour connaître à tout moment la position des moutons, on définit par SYMBOL les caractères graphiques.

Le compteur F donne le nombre de moutons déjà dans la bergerie, tandis que E compte le nombre d'essais à chaque partie.

Lignes 200 à 270 : dessin du jeu

Cette partie du programme affiche à l'écran la clôture du pâturage où se trouvent les moutons avec les caractères graphiques définis dans l'initialisation.

Lignes 300 à 450 : affiche berger, chien et moutons

Ces trois éléments sont affichés sur l'écran à des positions aléatoires. Les coordonnées du chien sont sauvegardées dans les variables X et Y, et celles des moutons dans les tableaux X() et Y(). C'est la variable N qui détermine le nombre de moutons (entre 5 et 9).

La bergerie est aussi déplacée à chaque partie. Ses coordonnées sont conservées dans les variables V et W.

Lignes 500 à 570 : déplacement du chien

La touche appuyée étant stockée dans la variable D$, on modifie en conséquence les variables X et Y si aucun élément sur l'écran ne s'y oppose (testé par TEST).

Si cette touche correspond à ENTER du clavier (code 13), on va en 580 pour le déplacement des moutons.

Lignes 580 à 830 : déplacement des moutons

Chaque mouton est déplacé à tour de rôle cinq fois de suite dans la direction qui l'éloigne le plus du chien.

S'il se trouve dans la bergerie (coordonnées V et W), on incrémente le compteur F. Le mouton ne peut alors plus en sortir.

Si le compteur F est égal à N, on va directement à la ligne 900 pour la fin du jeu.

Dans le cas contraire, on revient en ligne 500 pour un nouvel essai.



MISSIONNAIRES ET CANNIBALES

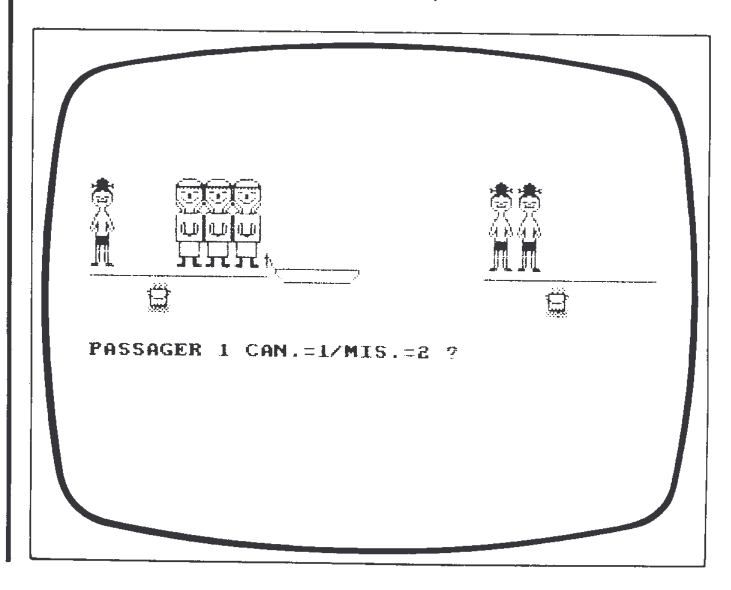
Ne vous fiez pas à la mine angélique des six personnages qui vont apparaître sur votre écran. En effet, les trois cannibales guettent le premier moment d'inattention des trois missionnaires pour les précipiter dans leurs marmites bouillantes.

Pourrez-vous les faire tous passer sur l'autre rive en sachant que les missionnaires ne doivent jamais être en minorité sur l'une des deux berges ?

Vous avez à votre disposition une barque qui ne peut prendre plus de deux passagers par traversée. A chaque voyage, l'ordinateur vous demande le nombre de cannibales ou de missionnaires à installer dans la barque (tapez 1 pour cannibale, 2 pour missionnaire et 0 s'il n'y a pas de deuxième passager). Puis la barque déposera ses voyageurs sur l'autre rive et devra en ramener au moins un sur la première.

Attention : dès que les cannibales deviennent plus nombreux que les missionnaires sur l'une des rives, ils entraînent automatiquement leurs prisonniers dans leurs marmites. Dans ce cas, tapez sur n'importe quelle touche et vos personnages ressusciteront pour un nouvel essai.

A vous de trouver la meilleure tactique.



```
10 REM MISSIONNAIRES & CANNIBALES
20 REM ---------------------------------
30 SYMBOL AFTER 128:GOSUB 900
40 C(1)=0:C(2)=12:C(3)=0:C(4)=12
50 P(1)=-1:P(2)=5:P(3)=27:P(4)=33
60 B=1:B(1)=14:B(2)=22
70 M(1)=3:M(2)=3
80 MODE 1:INK 0,1:INK 1,15:PAPER 0:PEN 1:BORDER 1
90 PLOT 0,150:DRAW 200,150
100 PLOT 442,150:DRAW 636,150
110 LOCATE 5,17:PRINT CHR$(157);CHR$(158);TAB(33);
CHR$(157);CHR$(158)
120 LOCATE 5,18:PRINT CHR$(159);CHR$(160);TAB(33);
CHR$(159);CHR$(160)
180 REM -------------------------------
190 REM LE JEU
200 LOCATE B(B),14:PRINT "      "
210 LOCATE B(B)-1,15:IF B=1 THEN PRINT CHR$(155);"
      " ELSE PRINT "      ";CHR$(156)
220 LOCATE B(B),16:PRINT CHR$(152);CHR$(153);CHR$(
153);CHR$(153);CHR$(153);CHR$(154)
230 GOSUB 700
240 IF M(3)+M(4)=6 THEN 500
250 IF (M(1)>M(2) AND M(2)<>0) OR (M(3) >M(4) AND
M(4)<>0) THEN 600
260 FOR I=1 TO 2
270 LOCATE 1,21:PRINT CHR$(18);"PASSAGER";I;"CAN.=
1/MIS.=2 ? ";
280 P$=INKEY$:IF P$="" THEN 280 ELSE P=VAL(P$):IF
P<0 OR P>2 THEN 280
290 IF I=1 AND P=0 THEN 280
300 IF P=0 THEN 380
310 IF B=2 THEN 340
320 IF M(P)=0 THEN 280
330 M(P)=M(P)-1:GOTO 360
340 IF M(B+P)=0 THEN 280
350 M(B+P)=M(B+P)-1
360 LOCATE B(B)-1+2*I,14:PRINT CHR$(C(P)+128);CHR$
(C(P)+129)
370 LOCATE B(B)-1+2*I,15:PRINT CHR$(C(P)+130);CHR$
(C(P)+131)
380 NEXT I
390 LOCATE 1,21:PRINT CHR$(18)
400 GOSUB 700
410 LOCATE B(B),14:PRINT "      ":LOCATE B(B)-1,1
5:PRINT "        ":LOCATE B(B),16:PRINT "      "
420 B=B+1:IF B=3 THEN B=1
430 IF B=2 THEN M(3)=3-M(1):M(4)=3-M(2):GOTO 200
440 M(1)=3-M(3):M(2)=3-M(4):GOTO 200
480 REM -------------------------------
490 REM GAGNE
500 LOCATE 4,20:PRINT"VOUS AVEZ ECHAPPE A LA MARMI
TE"
510 PRINT CHR$(7):LINE INPUT A$
520 RUN
```

```
580 REM ------------------------------
590 REM PERDU
600 LOCATE 4,20:PRINT "VOUS AVEZ FINI DANS LA MARM
ITE"
610 PRINT CHR$(7):LINE INPUT A$
620 RUN
680 REM ------------------------------
690 REM AFFICHE LE JEU
700 FOR I=1 TO 4
710 FOR J=1 TO 3
720 G=C(I)+128
730 FOR K=10 TO 15
740 IF J<=M(I) THEN LOCATE P(I)+J*2,K:PRINT CHR$(G
);CHR$(G+1):GOTO 760
750 LOCATE P(I)+J*2,K:PRINT "  "
760 G=G+2
770 NEXT K,J,I
780 RETURN
880 REM ------------------------------
890 REM GRAPHIQUES
900 SYMBOL 128,3,15,31,207,127,207,31,60
910 SYMBOL 129,0,192,224,204,248,204,224,240
920 SYMBOL 130,96,64,92,67,72,79,32,24
930 SYMBOL 131,24,8,232,8,72,200,16,96
940 SYMBOL 132,4,4,4,8,24,16,32,32
950 SYMBOL 133,128,128,128,64,96,32,16,16
960 SYMBOL 134,72,72,88,152,168,72,16,16
970 SYMBOL 135,72,72,104,100,84,72,32,32
980 SYMBOL 136,31,31,31,31,31,20,20,20
990 SYMBOL 137,224,224,224,224,224,160,160,160
1000 SYMBOL 138,20,20,20,20,20,20,120,248
1010 SYMBOL 139,160,160,160,160,160,160,120,124
1020 SYMBOL 140,15,48,64,192,255,170,128,156
1030 SYMBOL 141,240,12,2,3,255,171,1,57
1040 SYMBOL 142,129,129,145,136,68,163,144,72
1050 SYMBOL 143,129,129,137,17,34,197,9,18
1060 SYMBOL 144,36,20,52,79,128,128,128,128
1070 SYMBOL 145,36,40,44,242,1,1,1,1
1080 SYMBOL 146,140,148,148,148,148,148,143,129
1090 SYMBOL 147,49,41,41,41,41,41,241,129
1100 SYMBOL 148,128,128,255,32,32,32,32,32
1110 SYMBOL 149,1,1,255,4,4,4,4,4
1120 SYMBOL 150,32,32,63,12,12,12,124,124
1130 SYMBOL 151,4,4,252,48,48,48,62,62
1140 SYMBOL 152,255,128,128,64,32,16,8,7
1150 SYMBOL 153,255,0,0,0,0,0,0,255
1160 SYMBOL 154,255,1,1,2,4,8,16,224
1170 SYMBOL 155,16,16,56,20,20,18,18,17
1180 SYMBOL 156,2,2,7,10,10,18,18,34
1190 SYMBOL 157,9,4,9,63,224,160,96,48
1200 SYMBOL 158,32,144,32,240,28,20,24,48
1210 SYMBOL 159,47,32,32,42,191,85,42,85
1220 SYMBOL 160,208,16,16,176,244,168,80,168
1230 RETURN
```

PROGRAMME

Lignes 30 à 120 : initialisation

Après la définition des graphiques (sous-programme en 900), on définit les tableaux C, P, B et M pour connaître à tout moment la composition du jeu sur chaque berge (M), leurs positions sur l'écran (P), les caractères graphiques qui les composent (C) et la position sur l'écran de la barque (B).

Lignes 200 à 440 : le jeu

La variable B donnant le numéro de la berge (B=1 gauche, B=2 droite), on affiche la barque à B(B). Puis les personnages sont affichés par le sous-programme en 700 en fonction de M() et à la position P().

Dès la ligne 240, on vérifie s'ils sont tous sur la berge de droite. Si c'est le cas, on va en ligne 500 pour un message de félicitations.

La ligne 250, elle, teste si le nombre de cannibales n'est pas supérieur à celui des missionnaires. C'est en ligne 600 que l'on va, dans ce cas.

Puis, c'est la saisie des passagers effectuée par une boucle FOR/NEXT. On vérifie bien évidemment qu'il y ait au moins un passager (ligne 290) et qu'il existe bien sur la berge les voyageurs que l'on veut embarquer (lignes 310 à 350).

Ces vérifications étant faites, on affiche leurs têtes sur la barque (lignes 360 et 370).

Pour finir cette partie du programme, on affiche à nouveau les personnages (sous-programme en 700), on efface la barque et on réajuste le tableau M suivant le numéro de la berge.

Lignes 700 à 780 : affiche le jeu

En fonction des tableaux M, P et C, on affiche à l'écran les six personnages sur les deux berges. La variable G pointe le premier des caractères graphiques.

LISTE DES VARIABLES :

B	numéros de berges: B=1 berge gauche, B=2 berge droite
B()	position des berges sur l'écran
C()	numéros des caractères graphiques correspondant à M()
G	pointeur des caractères graphiques
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
K	indice de boucle FOR/NEXT
M()	position des missionnaires et des cannibales sur chaque berge
P$	dernière touche enfoncée
P	choix des passagers
P()	position de chaque personnage sur l'écran



PARITES

Dès que vous aurez rentré le nombre de joueurs (jusqu'à 5) et leurs prénoms, seize paires de figures diverses (cartes, téléphone, pièce de monnaie...) se dessinent faces cachées sur votre écran.

Elles se présentent sur quatre lignes de huit colonnes, le but du jeu étant de retrouver les seize paires mélangées.

Pour cela il faut retourner chacun à son tour deux cartes de façon à situer les positions des paires pour les reconstituer.

Amstrad demande les coordonnées des cartes sous forme du numéro de ligne, puis du numéro de colonne.

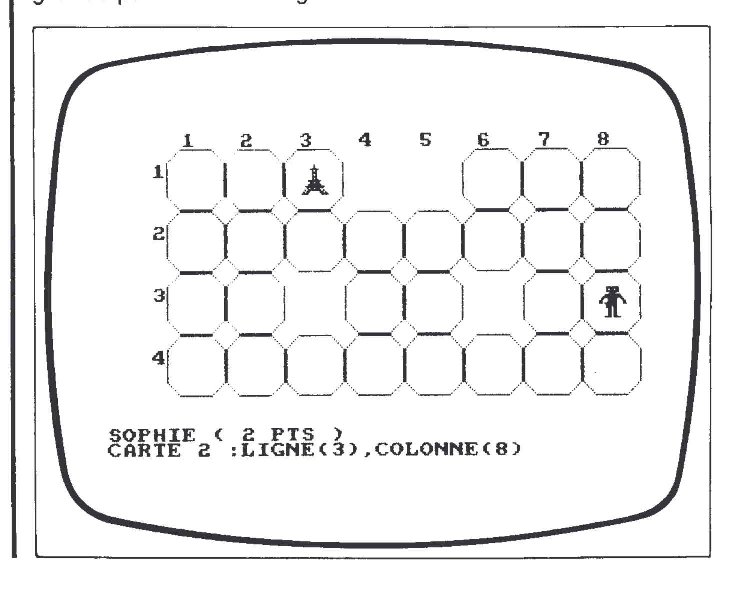
Il refuse les cartes déjà mises en paire et celles dont les coordonnées sont erronées.

Dans le cas où l'un des joueurs trouve une paire, les cartes sont effacées de l'écran.

Chaque paire trouvée fait obtenir un point et le gagnant peut rejouer.

Quand toute la grille est vide, les résultats de tous les joueurs s'affichent.

Ce jeu, apparemment simple, demande cependant une grande patience et une grande attention.



```
10 REM PARITES
20 REM ------------------------------
30 SYMBOL AFTER 128:GOSUB 1200
40 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1
50 NJ=0:TP=0
60 INPUT "NB DE JOUEURS ";JN
70 IF JN<2 OR JN>5 THEN PRINT"ENTRE 2 ET 5":GOTO 6
0
80 FOR I=1 TO JN
90 CLS:PRINT "NOM DU JOUEUR";I;:INPUT N$(I)
100 NEXT I
110 A$(1)=CHR$(196)+CHR$(193)+CHR$(193)+CHR$(197)
120 A$(2)=CHR$(195)+"  "+CHR$(194):A$(3)=A$(2)
130 A$(4)=CHR$(197)+CHR$(192)+CHR$(192)+CHR$(196)
180 REM ------------------------------
190 REM INITIALISATION
200 RANDOMIZE TIME
210 K=0
220 FOR I=1 TO 4
230 FOR J=1 TO 4
240 C(I,J)=K:C(I+4,J)=K
250 K=K+4
260 NEXT J,I
270 CLS:LOCATE 7,11:PRINT "MELANGE DES CARTES:"
280 FOR I=1 TO 120
290 L1=INT(RND*4)+1
300 L2=INT(RND*4)+1
310 C1=INT(RND*8)+1
320 C2=INT(RND*8)+1
330 A=C(C1,L1):C(C1,L1)=C(C2,L2):C(C2,L2)=A
340 LOCATE 26,11:PRINT 120-I;" "
350 NEXT I
380 REM ------------------------------
390 REM AFFICHE LES CARTES
400 CLS:PEN 3
410 FOR I=1 TO 8:PRINT TAB(1+I*4);I;:NEXT I
420 FOR J=0 TO 3
430 PEN 3:LOCATE 3,3+J*4:PRINT J+1
440 FOR I=0 TO 7
450 FOR K=1 TO 4
460 PEN 1:LOCATE 5+I*4,1+J*4+K:PRINT A$(K)
470 NEXT K,I,J
580 REM ------------------------------
590 REM LE JEU
600 NJ=NJ+1:IF NJ>JN THEN NJ=1
610 LOCATE 1,20:PRINT CHR$(18);N$(NJ);" (";P(NJ);"
PTS )
620 FOR I=1 TO 2
630 LOCATE 1,21:PRINT "CARTE";I;":LIGNE( ),COLONNE
( )"
```

```
640 A$=INKEY$:IF A$="" THEN 640 ELSE Y(I)=VAL(A$)
650 IF Y(I)<1 OR Y(I)>4 THEN 640
660 LOCATE 16,21:PRINT A$
670 A$=INKEY$:IF A$="" THEN 670 ELSE X(I)=VAL(A$)
680 IF X(I)<1 OR X(I)>8 THEN 670
690 LOCATE 27,21:PRINT A$
700 IF C(X(I),Y(I))=-1 THEN 630
710 IF I=2 AND X(1)=X(2) AND Y(1)=Y(2) THEN 630
720 C=C(X(I),Y(I))+128
730 PEN 2:LOCATE 2+X(I)*4,Y(I)*4-1:PRINT CHR$(C);C
HR$(C+2)
740 LOCATE 2+X(I)*4,Y(I)*4:PRINT CHR$(C+1);CHR$(C+
3):PEN 1
750 NEXT I
760 FOR I=1 TO 600:NEXT I
770 IF C(X(1),Y(1))=C(X(2),Y(2)) THEN 830
780 FOR I=1 TO 2
790 LOCATE 2+X(I)*4,Y(I)*4-1:PRINT "  "
800 LOCATE 2+X(I)*4,Y(I)*4:PRINT "  "
810 NEXT I
820 GOTO 600
830 P(NJ)=P(NJ)+1
840 FOR I=1 TO 2
850 FOR J=1 TO 4
860 LOCATE 1+X(I)*4,J+Y(I)*4-3:PRINT "    "
870 NEXT J,I
880 C(X(1),Y(1))=-1:C(X(2),Y(2))=-1:TP=TP+1
890 IF TP<16 THEN 610
980 REM ---------------------------
990 REM RESULTATS
1000 CLS
1010 LOCATE 14,5:PRINT "RESULTATS"
1020 LOCATE 5,8:PRINT "NOM";TAB(20);"POINTS"
1030 FOR I=1 TO JN
1040 LOCATE 5,9+I*2:PRINT N$(I);TAB(20);P(I)
1050 P(I)=0
1060 NEXT I
1070 LINE INPUT A$
1080 TP=0:GOTO 200
1180 REM ---------------------------
1190 REM GRAPHIQUES
1200 SYMBOL 128,7,5,7,1,15,31,51,99
1210 SYMBOL 129,111,7,7,3,3,3,3,15
1220 SYMBOL 130,240,176,240,192,248,252,230,227
1230 SYMBOL 131,227,224,96,96,96,96,96,120
1240 SYMBOL 132,0,0,7,4,4,31,16,19
1250 SYMBOL 133,19,16,16,19,18,18,16,31
1260 SYMBOL 134,0,192,224,32,34,250,10,202
1270 SYMBOL 135,202,10,10,206,72,72,8,248
1280 SYMBOL 136,0,3,13,16,32,32,64,97
1290 SYMBOL 137,65,33,32,16,13,3,0,0
```

```
1300 SYMBOL 138,0,128,96,16,56,72,132,12
1310 SYMBOL 139,4,8,8,16,96,128,0,0
1320 SYMBOL 140,0,3,12,16,33,35,69,65
1330 SYMBOL 141,65,65,33,33,16,12,3,0
1340 SYMBOL 142,0,192,48,8,4,116,66,98
1350 SYMBOL 143,66,66,4,4,8,48,192,0
1360 SYMBOL 144,0,63,196,159,228,9,18,36
1370 SYMBOL 145,40,40,36,34,33,32,63,24
1380 SYMBOL 146,0,252,35,249,39,144,72,36
1390 SYMBOL 147,20,20,100,100,148,4,252,24
1400 SYMBOL 148,127,64,84,92,72,64,64,66
1410 SYMBOL 149,67,65,64,84,92,72,64,127
1420 SYMBOL 150,252,4,84,116,36,4,4,132
1430 SYMBOL 151,132,4,4,84,116,36,4,252
1440 SYMBOL 152,255,68,68,255,17,17,255,68
1450 SYMBOL 153,68,255,17,17,255,68,68,255
1460 SYMBOL 154,255,68,68,255,17,17,255,68
1470 SYMBOL 155,68,255,17,17,255,68,68,255
1480 SYMBOL 156,63,32,36,32,32,36,32,63
1490 SYMBOL 157,32,32,32,32,36,32,63,0
1500 SYMBOL 158,252,4,36,4,4,36,4,252
1510 SYMBOl 159,4,36,4,4,4,4,252,0
1520 SYMBOL 160,3,6,10,18,34,74,130,146
1530 SYMBOL 161,131,164,137,144,164,192,255,0
1540 SYMBOL 162,255,1,73,1,73,1,73,1
1550 SYMBOL 163,255,2,36,8,144,32,192,0
1560 SYMBOL 164,1,64,200,48,25,141,95,35
1570 SYMBOL 165,83,139,11,3,3,3,3,7
1580 SYMBOL 166,32,162,68,136,52,227,128,241
1590 SYMBOL 167,202,164,146,168,128,128,128,192
1600 SYMBOL 168,0,0,12,30,63,127,127,63
1610 SYMBOL 169,30,12,0,0,0,0,0,0
1620 SYMBOL 170,7,26,100,132,8,136,144,16
1630 SYMBOL 171,24,60,126,255,255,126,60,24
1640 SYMBOL 172,0,0,3,1,1,1,1,1
1650 SYMBOL 173,7,3,7,31,11,23,44,120
1660 SYMBOL 174,128,128,224,64,192,64,192,64
1670 SYMBOL 175,240,96,240,252,104,244,26,15
1680 SYMBOL 176,63,96,192,135,140,152,248,152
1690 SYMBOL 177,24,48,96,192,255,1,3,3
1700 SYMBOL 178,252,6,3,225,49,25,31,25
1710 SYMBOL 179,24,12,6,3,255,128,192,192
1720 SYMBOL 180,0,1,19,31,255,127,63,31
1730 SYMBOL 181,31,31,31,63,63,127,31,0
1740 SYMBOL 182,192,224,252,255,254,254,254,252
1750 SYMBOL 183,252,252,254,254,255,252,128,0
1760 SYMBOL 184,0,0,0,0,0,7,31,127
1770 SYMBOL 185,127,255,255,255,255,255,127,62
1780 SYMBOL 186,0,7,24,44,206,199,131,129
1790 SYMBOL 187,128,192,192,192,128,128,0,0
1800 SYMBOL 188,1,7,15,63,255,80,76,69
```

```
1810 SYMBOL 189,65,65,73,68,35,17,8,7
1820 SYMBOL 190,0,192,224,248,254,20,102,70
1830 SYMBOL 191,6,6,38,70,138,21,39,197
1840 SYMBOL 192,0,0,0,0,0,0,0,255
1850 SYMBOL 193,255,0,0,0,0,0,0,0
1860 SYMBOL 194,1,1,1,1,1,1,1,1
1870 SYMBOL 195,128,128,128,128,128,128,128,128
1880 SYMBOL 196,1,2,4,8,16,32,64,128
1890 SYMBOL 197,128,64,32,16,8,4,2,1
1900 RETURN
```

Liste des variables :

A$	dernière touche appuyée
A$( )	dessin des cartes
C	pointeur des caractères graphiques pour l'affichage
C(,)	tableau des cartes
C1 et **C2**	nombres aléatoires pour le mélange des cartes
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
JN	nombre de joueurs
K	indice de boucle FOR/NEXT
L1 et **L2**	nombres aléatoires pour le mélange des cartes
N$()	nom des joueurs
NJ	numéro du joueur
P()	total des points de chaque joueur
TP	total des points
X() et **Y()**	coordonnées des cartes choisies

Programme

Lignes 30 à 350 : initialisation

Après la définition des caractères graphiques (sous-programme en 1200), on saisit au clavier le nombre et le nom des joueurs et on place dans le tableau A$( ) le dessin des cartes.

Le numéro des caractères graphiques de chaque carte étant placé dans le tableau C(,), on mélange ce dernier en utilisant les quatre nombres aléatoires C1, C2, L1 et L2.

Lignes 400 à 470 : affichage des cartes

On affiche à l'écran les 32 cartes avec les caractères contenus dans le tableau A$.

Lignes 600 à 820 : le jeu

Après la saisie dans la variable A$, les numéros de lignes et de colonnes sont placés respectivement dans les tableaux Y() et X().

Dans le cas où ces deux cartes sont identiques, on saute directement à la ligne 830; sinon, on les retourne après une petite boucle d'attente.

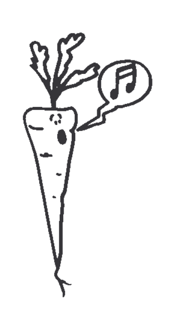
Lignes 830 à 890 : cas des cartes identiques

On incrémente le total des points du joueur (tableau P), puis l'on efface de l'écran les cartes désignées.

Si le total des cartes effacées est inférieur à 16, on continue normalement le jeu mais en gardant le même joueur (ligne 610 au lieu de 600).

Lignes 1000 à 1080 : résultats

On affiche le nom et le nombre de points de chaque joueur (tableaux N$ et P).



CHAMP DE CAROTTES

C'est un jour d'hiver bien froid et maître lapin ne voit plus qu'une carotte dans le fond du champ. Il va essayer d'aller la déterrer, tout en sachant que les villageois ont tendu des pièges sous l'herbe.

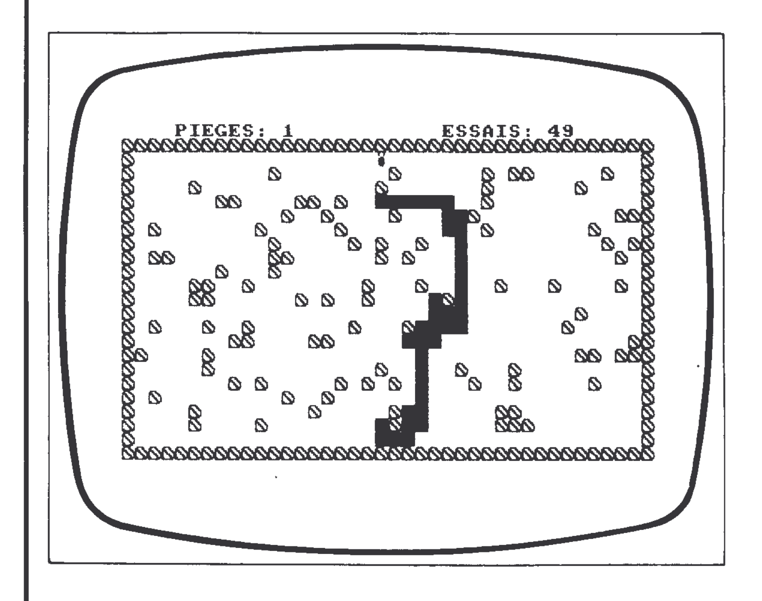
Heureusement qu'il porte sur lui son fidèle détecteur électronique...

Le détecteur inscrit en haut à gauche de l'écran le nombre des pièges qui entourent le lapin, en comptant aussi tous ceux qui composent la bordure du champ.

Vous n'avez plus qu'à guider logiquement votre lapin en utilisant les quatre touches de déplacement.

Que vous ayez évité ou non les pièges, vous découvrirez à chaque fin de jeu leur plan précis avec votre nombre d'essais parfois impressionnant.

Si votre lapin, toujours affamé, veut repartir à la recherche d'une autre carotte, appuyez sur ENTER.



```
10 REM CHAMP DE CAROTTES
20 REM --------------------------------
30 DIM E(40,25):SYMBOL AFTER 128:GOSUB 800
40 X(1)=1:X(2)=-1
50 Y(3)=1:Y(4)=-1
60 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,8:PAPER 0
:PEN 1:BORDER 1
70 RANDOMIZE TIME
80 FOR I=1 TO 100
90 X=INT(RND*38)+2
100 Y=INT(RND*19)+4
110 IF E(X,Y)=1 THEN 90
120 E(X,Y)=1
130 NEXT I
140 PEN 1
180 REM --------------------------------
190 REM AFFICHE LE JEU
200 FOR I=2 TO 40
210 LOCATE I,2:PRINT CHR$(129):LOCATE I,24:PRINT C
HR$(129)
220 IF I<25 THEN LOCATE 1,I:PRINT CHR$(129):LOCATE
 40,I:PRINT CHR$(129)
230 NEXT I
240 PEN 1:LOCATE 20,3:PRINT CHR$(130):PEN 2:LOCATE
 20,23:PRINT CHR$(128)
250 X=20:Y=23
260 E=0
280 REM --------------------------------
290 REM NB DE PIEGES
300 P=0
310 FOR I=1 TO 4
320 IF E(X+X(I),Y+Y(I))=1 OR TEST((X+X(I))*16-8,40
6-(Y+Y(I))*16)=1 THEN P=P+1
330 NEXT I
340 LOCATE 5,1:PEN 2:PRINT "PIEGES:";:PEN 3:PRINT
P;
350 PEN 2:PRINT TAB(25);"ESSAIS:";:PEN 3:PRINT E:P
EN 2
380 REM --------------------------------
390 REM DEPLACEMENTS
400 A$=INKEY$:IF A$="" THEN 400
410 IF A$=CHR$(243) THEN D=1:GOTO 460
420 IF A$=CHR$(242) THEN D=2:GOTO 460
430 IF A$=CHR$(241) THEN D=3:GOTO 460
440 IF A$=CHR$(240) THEN D=4:GOTO 460
450 GOTO 400
460 PAPER 3:LOCATE X,Y:PRINT " ":PAPER 0
470 X=X+X(D):Y=Y+Y(D)
480 IF E(X,Y)=1 OR TEST(X*16-8,406-Y*16)=1 THEN 60
0
490 LOCATE X,Y:PRINT CHR$(128)
```

```
500 E=E+1
510 GOTO 300
580 REM ------------------------------
590 REM FIN DE PARTIE
600 IF X=20 AND Y=3 THEN FOR I=500 TO 50 STEP -5:S
OUND 1,I,2:NEXT:GOTO 640
610 FOR I=0 TO 15
620 PRINT CHR$(7);:BORDER I:FOR J=1 TO 60:NEXT
630 NEXT I
640 FOR I=1 TO 39
650 FOR J=3 TO 22
660 IF E(I,J)=1 THEN LOCATE I,J:PRINT CHR$(129)
670 NEXT J,I
680 LINE INPUT A$
690 RUN
780 REM ------------------------------
790 REM GRAPHIQUES
800 SYMBOL 128,231,36,60,126,90,126,36,24
810 SYMBOL 129,120,204,230,179,153,141,199,254
820 SYMBOL 130,36,36,24,60,60,60,60,24
830 RETURN
```

PROGRAMME

Lignes 30 à 140 : initialisation

On place dans les tableaux X() et Y() le nombre de caractères de déplacement correspondant aux quatre · directions.

Puis on place dans le tableau E les pièges à des positions aléatoires (100 pièges au total).

Lignes 200 à 260 : affiche le jeu

Cette partie du programme affiche simplement des pièges au bord de l'écran (visibles cette fois) ainsi que le lapin et la carotte.

Les variables X et Y donnent la position exacte du lapin.

Lignes 300 à 350 : nombre de pièges

Pour connaître le nombre de pièges autour du lapin, il suffit de tester le tableau E à l'aide des tableaux X et Y.

Lignes 400 à 510 : déplacements

Le caractère saisi au clavier étant stocké dans la variable A$, on modifie suivant sa valeur la variable D.

Avant d'afficher le lapin, on place un carré violet sur son ancienne position.

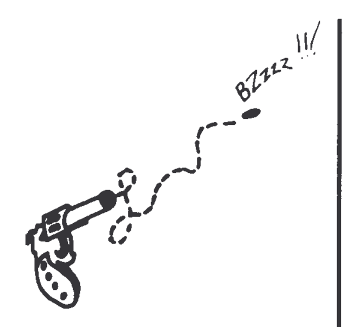
La ligne 480 teste si le lapin n'est pas sur un piège ou sur la carotte.

Lignes 600 à 690 : fin de partie

Quelle que soit l'issue du jeu, on affiche à l'écran tous les pièges jusqu'alors invisibles en testant le tableau E.

LISTE DES VARIABLES :

A$	caractère saisi au clavier
D	direction du déplacement
E	nombre d'essais
E()	position des pièges sur l'écran
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
P	nombre de pièges autour du lapin
X et **Y**	coordonnées du lapin
X() et **Y()**	nombre de caractères de déplacement dans les 4 directions.



BALISTIQUE

Prenez deux voisins qui ne s'entendent pas, leurs maisons et le mur qui les sépare.

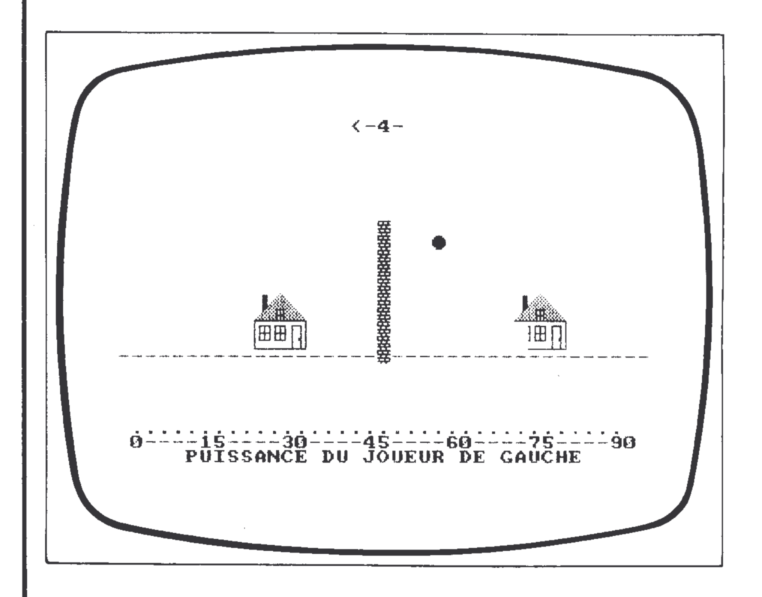
Ils n'ont de cesse de s'envoyer réciproquement des projectiles afin de détruire l'habitation adverse.

Selon les lois de la civilité, ils tirent chacun à leur tour, en réglant soigneusement leurs angle et puissance (de 0 à 90) d'après la force et le sens du vent qui souffle ce jour-là, et l'éloignement de la cible.

Les deux joueurs qui les représentent se servent des flèches gauche et droite et valident leurs coordonnées en appuyant sur ENTER.

Le premier des deux adversaires qui détruit au moins 10 éléments d'une maison est considéré comme gagnant.

Qu'à cela ne tienne : la revanche ne va pas tarder !



```
10 REM BALISTIQUE
20 REM -----------------------------
30 SYMBOL AFTER 128:GOSUB 1300
40 RANDOMIZE TIME
50 MODE 1:INK 0,11:INK 1,24:INK 2,14:INK 3,15:PAPE
R 2:PEN 1:BORDER 11
60 X=INT(RND*12)+1
70 P(1)=X+5
80 GOSUB 1000
90 X=INT(RND*12)+24
100 P(2)=X-1
110 GOSUB 1000
120 PAPER 0:PEN 3:PRINT:PRINT STRING$(39,"-")
130 FOR I=8 TO 17:LOCATE 20,I:PRINT CHR$(144):NEXT
 I:PEN 1
140 VV=INT(RND*5)+1
150 SV=INT(RND+0.5)
160 V(1)=VV:V(2)=VV:PEN 1
170 IF SV=0 THEN V(1)=-V(1):LOCATE 18,1:PRINT "<-"
;CHR$(VV+48);"-":GOTO 190
180 V(2)=-V(2):LOCATE 19,1:PRINT"-";CHR$(VV+48);"-
>"
190 PEN 1:S(1)=1:S(2)=-1
200 A=1.74532925E-02
210 J$(1)="GAUCHE":J$(2)="DROITE"
220 S$(1)="ANGLE":S$(2)="PUISSANCE"
230 LOCATE 2,22
240 FOR I=1 TO 36:PRINT".";:NEXT I
250 PRINT:PRINT" 0----";
260 FOR I=15 TO 90 STEP 15
270 PRINT MID$(STR$(I),2,2);"----";
280 NEXT I:LOCATE 39,23:PRINT "     "
380 REM -----------------------------
390 REM LE JEU
400 IF J=1 THEN J=2 ELSE J=1
410 GOSUB 1100
420 R(1)=R(1)*3
430 IX=COS(R(1)*A)
440 IY=SIN(R(1)*A)
450 X=P(J):Y=12
460 FOR I=1 TO R(2)+V(J)
470 FOR K=1 TO 30:NEXT:LOCATE INT(X),INT(Y):PRINT
" "
480 X=X+IX*S(J)
490 Y=Y-IY
500 IF X<2 OR X>39 OR Y<2 OR Y>17 THEN 400
510 Z=TEST(INT(X)*16-8,406-INT(Y)*16)
520 IF Z<>0 THEN 800
530 LOCATE INT(X),INT(Y):PRINT CHR$(145)
540 NEXT I
550 FOR I=1 TO (R(2)-V(J)*S(J))/2
```

```
560 FOR K=1 TO 30:NEXT:LOCATE INT(X),INT(Y):PRINT
" "
570 X=X+IX*S(J)
580 IF X<2 OR X>39 OR Y<2 OR Y>17 THEN 400
590 Z=TEST(INT(X)*16-8,406-INT(Y)*16)
600 IF Z<>0 THEN 800
610 LOCATE INT(X),INT(Y):PRINT CHR$(145)
620 NEXT I
630 FOR K=1 TO 30:NEXT:LOCATE INT(X),INT(Y):PRINT
" "
640 X=X+IX*S(J)
650 Y=Y+IY
660 IF X<2 OR X>39 OR Y<2 OR Y>17 THEN 400
670 Z=TEST(INT(X)*16-8,406-INT(Y)*16)
680 IF Z<>0 THEN 800
690 LOCATE INT(X),INT(Y):PRINT CHR$(145)
700 GOTO 630
780 REM ------------------------------
790 REM FIN DU TIR
800 IF TEST(INT(X)*16-8,408-INT(Y)*16)=3 THEN 400
810 IF Z<>2 AND Z<>1 THEN 400
820 LOCATE INT(X),INT(Y):PRINT " "
830 G(J)=G(J)+1:LOCATE 1+(J-1)*30,20:PRINT "PTS:";
G(J)
840 IF G(J)<10 THEN 400
850 LOCATE 6,24:PRINT " LE JOUEUR DE ";J$(J);" GAG
NE        ";
860 IF INKEY$<>CHR$(13) THEN 860
870 RUN
980 REM ------------------------------
990 REM AFFICHE LES MAISONS
1000 FOR I=0 TO 15
1010 IF I/4=INT(I/4) THEN LOCATE X,13+I/4
1020 PRINT CHR$(128+I);
1030 NEXT I
1040 RETURN
1080 REM ------------------------------
1090 REM SAISIES DES DONNEES
1100 FOR I=1 TO 2
1110 LOCATE 6,24:PRINT S$(I);" DU JOUEUR DE ";J$(J
);"          ";
1120 Q=20
1130 PAPER 1:PEN 0:LOCATE Q,22:PRINT "."
1140 A$=INKEY$:IF A$="" THEN 1140
1150 IF A$=CHR$(13) THEN 1190
1160 IF A$=CHR$(242) AND Q>2 THEN PAPER 0:PEN 1:LO
CATE Q,22:PRINT ".":Q=Q-1:GOTO 1130
1170 IF A$=CHR$(243) AND Q<37 THEN PAPER 0:PEN 1:L
OCATE Q,22:PRINT ".":Q=Q+1
1180 GOTO 1130
1190 R(I)=Q
```

```
1200 PAPER 0:PEN 1:LOCATE Q,22:PRINT "."
1210 NEXT I
1220 R(2)=R(2)/2
1230 RETURN
1280 REM ------------------------------
1290 REM GRAPHIQUES
1300 SYMBOL 128,0,7,7,7,7,7,7,7
1310 SYMBOL 129,1,2,5,10,21,42,85,170
1320 SYMBOL 130,0,128,64,160,80,168,84,170
1330 SYMBOL 131,0,0,0,0,0,0,0,0
1340 SYMBOL 132,7,6,5,10,21,42,85,255
1350 SYMBOL 133,85,175,85,173,87,173,85,255
1360 SYMBOL 134,85,234,85,106,213,106,85,255
1370 SYMBOL 135,0,128,64,160,80,168,84,254
1380 SYMBOL 136,128,128,159,146,146,146,159,146
1390 SYMBOL 137,0,0,207,73,73,73,207,73
1400 SYMBOL 138,0,0,227,34,34,34,226,34
1410 SYMBOL 139,2,2,242,18,18,18,18,18
1420 SYMBOL 140,146,146,159,128,128,128,128,255
1430 SYMBOL 141,73,73,207,0,0,0,0,255
1440 SYMBOL 142,34,34,226,2,2,2,2,255
1450 SYMBOL 143,18,50,50,18,18,18,18,254
1460 SYMBOL 144,255,146,255,36,255,73,255,36
1470 SYMBOL 145,60,126,255,255,255,255,126,60
1480 RETURN
```

LISTE DES VARIABLES :

A$	dernière touche enfoncée
A	conversion degrés-radians
G()	points de chaque joueur
I	indice de boucle FOR/NEXT
IX et **IY**	incréments horizontal et vertical
J	numéro du joueur
J$()	nom des joueurs
P()	position des maisons
Q	pointeur sur l'échelle
R()	réponse angle et puissance
S()	type de donnée à saisir
S$()	nom des données à saisir
SV	sens du vent
V()	vitesse et sens du vent pour chaque joueur
VV	vitesse du vent
X et **Y**	coordonnées de la pierre
Z	code du caractère sous la pierre

PROGRAMME

Lignes 30 à 280 : initialisation

Après la définition des caractères graphiques par le sous-programme en 1300, on dessine sur l'écran les deux maisons en utilisant le sous-programme en 1000 qui les affiche à la position X.

Dans les lignes 140 à 220, on définit les principales variables utilisées dans le programme (voir liste des variables).

Lignes 400 à 700 : le jeu

Les incréments verticaux et horizontaux du déplacement de la pierre sont calculés à partir de la variable A qui fait la conversion degrés-radians.

Le projectile est déplacé en trois temps : la montée (lignes 460 à 540), l'horizontale (lignes 550 à 620) et la descente (lignes 630 à 700). Celles-ci utilisent tous les paramètres déjà calculés : incréments, sens et vitesse du vent...

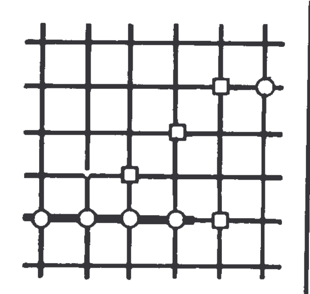
On teste à chaque déplacement qu'aucun obstacle ne se trouve sur son chemin.

Lignes 800 à 870 : fin du tir

Si l'une des parties encore visible sur l'écran est touchée, on incrémente le compteur de points du joueur. Tant que celui-ci est inférieur à 10, on continue le jeu.

Lignes 1100 à 1230 : saisie des données

Ce sous-programme saisit l'angle et la puissance donnés au projectile par chaque joueur. Suivant la valeur du caractère saisi dans A$, on déplace le point vers la droite ou vers la gauche sur l'échelle du bas de l'écran. Quand il s'agit de ENTER (code 13), on garde la dernière valeur de la variable Q.



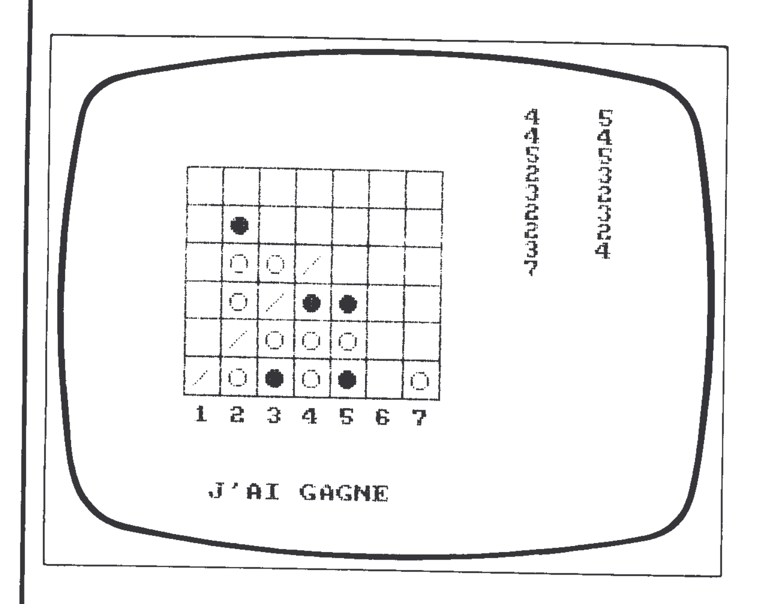
PUISSANCE 4

Vous connaissez sûrement le jeu « Puissance 4 » qu'Amstrad vous propose de jouer contre lui.

Il trace d'abord sur l'écran une grille de 7 cases sur 6 et attend que vous tapiez le numéro de la colonne où tombera votre pion (un chiffre de 1 à 7). Puis c'est à son tour de jouer, et ainsi de suite. Les coups des deux joueurs s'affichent à droite de l'écran.

Le joueur qui réussit à aligner quatre pions verticalement, horizontalement, ou en diagonale, a gagné la partie.

Il suffit de taper sur n'importe quelle touche pour obtenir une nouvelle grille.



```
10 REM PUISSANCE 4
20 REM --------------------------------
30 SYMBOL AFTER 128:DEFINT A-Z:GOSUB 1100
40 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1
50 FOR I=200 TO 424 STEP 32
60 PLOT I,344:DRAW 424,344:DRAW 424,152:DRAW I,152
:DRAW I,344
70 IF I<392 THEN PLOT 200,544-I:DRAW 424,544-I:DRA
W 424,248:DRAW 200,248:DRAW 200,544-I
80 NEXT I
90 PEN 3:LOCATE 14,17:PRINT "1 2 3 4 5 6 7"
100 DIM P(8,7)
110 E=1
120 X(1)=0:Y(1)=-1:X(2)=-1:Y(2)=-1
130 X(3)=1:Y(3)=-1:X(4)=-1:Y(4)=0
180 REM -------------------------------
190 REM LE JEU
200 LOCATE 1,21:PRINT CHR$(18);
210 IF E=2 THEN 400
220 PEN 1:LOCATE 13,21:PRINT"No DE COLONNE ?";
230 A$=INKEY$:IF A$="" THEN 230 ELSE A=VAL(A$)
240 IF A<1 OR A>7 OR P(A,1)<>0 THEN 230
250 GOSUB 780:J=A:C(J)=I:Z=E:GOSUB 600:P(A,I)=E
260 PEN 2
270 FOR J=0 TO I
280 LOCATE 12+A*2,3+(J-1)*2:PRINT " "
290 LOCATE 12+A*2,3+J*2:PRINT CHR$(127+E)
300 FOR K=1 TO 300:NEXT K
310 NEXT J:IF N>2 THEN 900
320 PEN 3:LOCATE 27+E*4,1+INT(T!):PRINT A:T!=T!+0.
5
330 IF T!=21 THEN 1000
340 E=E+1:IF E=3 THEN E=1
350 GOTO 200
380 REM -------------------------------
390 REM COUP DE L'ORDINATEUR
400 FOR J=1 TO 7
410 A=J:GOSUB 780:C(J)=I:IF I=0 THEN C(J)=-2:G(J)=
-1:GOTO 440
420 Z=1:GOSUB 600:G(J)=N:Q(J)=0
430 Z=2:GOSUB 600:H(J)=N:L(J)=0:IF N>2 THEN A=J:GO
TO 250
440 NEXT J:F=0
450 A=1:FOR J=1 TO 7:IF F=0 AND C(J)<0 THEN 490
460 IF F=1 AND C(J)<-1 THEN 490
470 IF G(J)>G(A) THEN A=J:GOTO 490
480 IF G(J)=G(A) AND L(J)>=L(A) AND Q(J)>=Q(A) THE
N A=J
490 NEXT J
500 GOSUB 780:P(A,I)=E:J=A:IF C(J)-1=0 THEN P(A,I)
=0:GOTO 530
```

```
510 C(J)=C(J)-1:Z=1:GOSUB 600:P(A,I)=0:C(J)=C(J)+1
520 IF N>2 THEN C(J)=-1:G(J)=-1:GOTO 450
530 IF F=0 AND C(J)<0 THEN F=1:GOTO 450
540 GOTO 250
580 REM -----------------------------
590 REM NB DE BLOCAGES
600 N=0:O=0
610 FOR K=1 TO 4
620 M=0:V=1
630 X=J:Y=C(J)
640 X=X+V*X(K):Y=Y+V*Y(K)
650 IF X<1 OR X>7 OR Y<1 OR Y>6 THEN 680
660 IF P(X,Y)<>Z THEN 680
670 M=M+1:O=O+1:GOTO 640
680 IF V=1 THEN V=-1:GOTO 630
690 IF M>N THEN N=M
700 IF M>=3 THEN W=K
710 NEXT K
720 RETURN
780 REM -----------------------------
790 REM POSITION LIGNE
800 FOR I=1 TO 6
810 IF P(A,I)<>0 THEN 830
820 NEXT I
830 I=I-1
840 RETURN
880 REM -----------------------------
890 REM FIN DE PARTIE
900 IF E=1 THEN LOCATE 13,21:PRINT "VOUS AVEZ GAGN
E":GOTO 920
910 LOCATE 15,21:PRINT "J'AI GAGNE"
920 V=1
930 X=A:Y=I
940 LOCATE 12+X*2,3+Y*2:PRINT CHR$(W+129)
950 X=X+V*X(W):Y=Y+V*Y(W)
960 IF P(X,Y)=E THEN 940
970 IF V=1 THEN V=-1:GOTO 930
980 IF INKEY$="" THEN 980
990 RUN
1000 LOCATE 11,21:PRINT "PARTIE NULLE"
1010 GOTO 980
1080 REM -----------------------------
1090 REM GRAPHIQUES
1100 SYMBOL 128,60,66,129,129,129,129,66,60
1110 SYMBOL 129,60,126,255,255,255,255,126,60
1120 SYMBOL 130,16,16,16,16,16,16,16,16
1130 SYMBOL 131,128,64,32,16,8,4,2,1
1140 SYMBOL 132,1,2,4,8,16,32,64,128
1150 SYMBOL 133,0,0,0,0,255,0,0,0
1160 RETURN
```

Programme

Lignes 30 à 130 : initialisation

Après la définition des caractères graphiques (sous-programme en 1100), les lignes 50 à 90 dessinent le damier.

On place dans les tableaux X et Y le nombre de cases de déplacement pour les recherches de l'ordinateur. On obtient les huit directions en multipliant par moins un les quatre valeurs des tableaux X et Y.

Lignes 200 à 350 : le jeu

La variable E détermine le tour du joueur (E=1) ou le tour de l'ordinateur (E=2).

Les lignes 220 à 240 saisissent au clavier le choix du joueur et vérifient sa réponse.

Si celle-ci est correcte, on va au sous-programme en 780 pour connaître la position du pion, et au sous-programme en 600 pour savoir si quatre pions ont été alignés.

Si la variable N est supérieure à 2 (au moins trois pions plus celui que l'on vient de jouer), on va en ligne 900 pour la fin de partie.

Les lignes 260 à 320 affichent le pion dans la grille ainsi que le coup qui vient d'être joué.

On modifie alors la valeur de la variable E avant de retourner au joueur suivant en ligne 200.

Lignes 400 à 540 : coup de l'ordinateur

L'ordinateur teste toutes les colonnes pour trouver la plus intéressante (soit pour bloquer l'adversaire, soit pour gagner).

Il utilise pour cela le sous-programme en 780 pour connaître l'emplacement d'un pion sur une colonne, ainsi que le sous-programme en 600 pour compter le nombre de pions alignés par lui ou par le joueur.

Lignes 600 à 720 : nombre de blocages

Ce sous-programme place dans la variable N le nombre maximum de pions alignés si l'on jouait dans la colonne J et pour les pions de type Z (Z=1 pion du joueur, Z=2 pions de l'ordinateur). Il effectue la même chose pour la variable O mais pour tous les pions (alignés ou non). Cette variable O est utilisée comme coefficient si aucune colonne n'est intéressante.

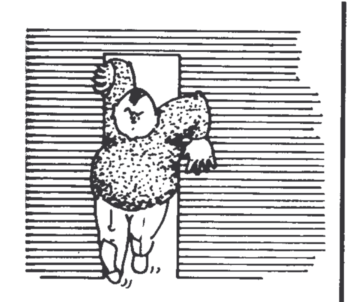
La variable V qui passe de 1 à −1 permet d'étudier les huit directions avec les tableaux X et Y.

Lignes 780 à 840 : position ligne

Ce programme place dans la variable I le premier numéro de ligne libre de la colonne désignée par la variable A.

LISTE DES VARIABLES :

A	valeur numérique de A$
A$	colonne choisie par le joueur
C()	conserve les numéros de ligne donnés par le sous-programme en 780
E	numéro du joueur (E=1 pour le joueur, E=2 pour Amstrad)
F	drapeau
G()	conserve les valeurs de la variable N pour Z=1
H()	conserve les valeurs de la variable N pour Z=2
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
K	indice de boucle FOR/NEXT
L()	conserve les valeurs de la variable O pour Z=2
M	nombre de pions alignés
N	nombre maximum de pions alignés autour de celui que l'on veut jouer
O	nombre de pions autour de celui que l'on veut jouer
P(,)	emplacement de tous les pions sur la grille
Q()	conserve les valeurs de la variable O pour Z=1
T!	nombre de demi-coups déjà joués
V	permet d'obtenir les huit directions avec les tableaux X et Y
W	direction dans laquelle les pions gagnants sont alignés
X et **Y**	coordonnées des pions gagnants
X() et **Y()**	nombre de cases de déplacement pour les quatre directions
Z	type de pion à analyser



BLOCAGES

Votre Amstrad vous demande tout de suite de donner le niveau de difficulté du jeu (entre 1 et 3). Après votre choix, il dessine à l'écran une grille de huit cases sur sept, puis le pion rouge du joueur et le pion orangé d'Amstrad.

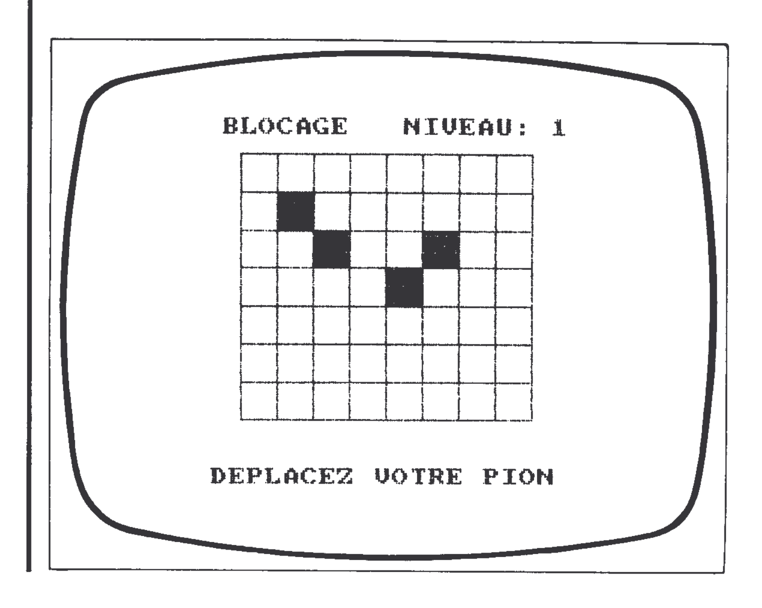
Le jeu peut alors commencer. Un coup se déroule en deux phases :

– vous déplacez d'une case et d'une seule votre pion à l'aide des quatre flèches de direction.

– puis vous manœuvrez avec ces mêmes flèches le curseur clignotant, passant s'il le faut, sur des cases déjà occupées. Vous validez à ce moment la case choisie pour bloquer votre adversaire en tapant sur ENTER. Quand vous jouez vous ne pouvez bloquer l'ordinateur que d'une case.

Par contre, quand ce sera le tour d'Amstrad, il vous bloquera de une à trois cases à la fois suivant le niveau du jeu que vous avez choisi en début de partie.

Ne perdez pas courage, il y a une stratégie pour bloquer votre adversaire même au niveau trois !



```
10 REM BLOCAGE
20 REM ------------------------------
30 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1
40 RANDOMIZE TIME
50 PRINT TAB(15);"BLOCAGE"
60 PRINT:PRINT:PRINT:PRINT
70 PRINT"  CHOISISSEZ VOTRE NIVEAU :":PRINT:PRINT
80 PRINT:PRINT"  1- FACILE"
90 PRINT:PRINT"  2- PAS EVIDENT"
100 PRINT:PRINT"  3- VRAIMENT TRES DIFFICILE"
110 LOCATE 15,21:PRINT"?";
120 NIV=VAL(INKEY$):IF NIV<1 OR NIV>3 THEN 120
180 REM ------------------------------
190 REM DESSIN DU JEU
200 CLS:PRINT TAB(11);"BLOCAGE   NIVEAU:";NIV
210 FOR X=1 TO 8:FOR Y=1 TO 7:PRINT CHR$(7);
220 PLOT X*32+144,400-Y*32:DRAWR 32,0:DRAWR 0,-32:
DRAWR -32,0:DRAWR 0,32
230 NEXT Y,X
240 XJ=3:YJ=4:XO=6:YO=4
250 X=XJ:Y=YJ:C=2:GOSUB 1010
260 X=XO:Y=YO:C=3:GOSUB 1010
380 REM ------------------------------
390 REM COUP DU JOUEUR
400 REM DEPLACEMENT
410 XX=XJ:YY=YJ:GOSUB 1060:IF T=0 THEN PAPER 0:LOC
ATE 1,23:PRINT CHR$(18);TAB(12);"VOUS AVEZ PERDU !
":GOSUB 1040:WHILE INKEY$="":WEND:RUN
420 PAPER 0:LOCATE 1,23:PRINT CHR$(18);TAB(11);"DE
PLACEZ VOTRE PION"
430 D$=INKEY$:IF D$="" THEN 430 ELSE D=ASC(D$):AX=
XJ:AY=YJ:X=XJ:Y=YJ
440 IF D=242 THEN IF X>1 THEN X=X-1
450 IF D=243 THEN IF X<8 THEN X=X+1
460 IF D=240 THEN IF Y>1 THEN Y=Y-1
470 IF D=241 THEN IF Y<7 THEN Y=Y+1
480 GOSUB 1050:IF P<>0 THEN GOSUB 1030:GOTO 430
490 C=2:GOSUB 1010:GOSUB 1020:XJ=X:YJ=Y
500 REM BLOCAGE D'UNE CASE
510 PAPER 0:LOCATE 1,23:PRINT CHR$(18);TAB(12);"BL
OQUEZ UNE CASE"
520 X=4:Y=4
530 GOSUB 1050:CC=P
540 C=1:GOSUB 1010:D$=INKEY$:C=0:GOSUB 1010:IF D$=
"" THEN 540
550 C=CC:GOSUB 1010:D=ASC(D$)
560 IF D=242 THEN IF X>1 THEN X=X-1
570 IF D=243 THEN IF X<8 THEN X=X+1
580 IF D=240 THEN IF Y>1 THEN Y=Y-1
590 IF D=241 THEN IF Y<7 THEN Y=Y+1
600 IF D<>13 THEN 530
610 IF CC<>0 THEN GOSUB 1030:GOTO 540
```

```
620 GOSUB 1000
680 REM -------------------------------
690 REM COUP DE L'ORDINATEUR
700 REM DEPLACEMENT
710 PAPER 0:LOCATE 1,23:PRINT CHR$(18)
720 XX=XO:YY=YO:GOSUB 1060:IF T=0 THEN PAPER 0:LOC
ATE 1,23:PRINT CHR$(18);TAB(14);"J'AI PERDU !":GOS
UB 1040:WHILE INKEY$="":WEND:RUN
730 AX=XO:AY=YO
740 IF XO>1 THEN X=XO-1:Y=YO:GOSUB 1050:IF P=0 THE
N IF INT(RND*9)=4 THEN 790
750 IF XO<8 THEN X=XO+1:Y=YO:GOSUB 1050:IF P=0 THE
N IF INT(RND*8)=4 THEN 790
760 IF YO>1 THEN X=XO:Y=YO-1:GOSUB 1050:IF P=0 THE
N IF INT(RND*7)=4 THEN 790
770 IF YO<7 THEN X=XO:Y=YO+1:GOSUB 1050:IF P=0 THE
N IF INT(RND*6)=4 THEN 790
780 GOTO 740
790 C=3:GOSUB 1010:PRINT CHR$(7);:GOSUB 1020:XO=X:
YO=Y
800 REM BLOCAGE DES CASES
810 FOR Z=1 TO NIV
820 FOR X=XJ-1 TO XJ+1:FOR Y=YJ-1 TO YJ+1
830 IF X<1 OR X>8 OR Y<1 OR Y>7 THEN 850
840 GOSUB 1050:IF P=0 THEN IF INT(RND*7)=4 THEN 86
0
850 NEXT Y,X:XX=XJ:YY=YJ:GOSUB 1060:IF T>0 THEN 82
0
860 PRINT CHR$(7);:GOSUB 1000
870 NEXT Z
880 GOTO 410
980 REM -------------------------------
990 REM ROUTINES D'AFFICHAGE
1000 PAPER 1:LOCATE X*2+10,Y*2+1:PRINT"  ":LOCATE
X*2+10,Y*2+2:PRINT"  ":RETURN
1010 PAPER C:LOCATE X*2+10,Y*2+1:PRINT"  ":LOCATE
X*2+10,Y*2+2:PRINT"  ":PLOT X*32+144,400-Y*32:DRAW
R 32,0:DRAWR 0,-32:DRAWR -32,0:DRAWR 0,32:RETURN
1020 PAPER 0:LOCATE AX*2+10,AY*2+1:PRINT"  ":LOCAT
E AX*2+10,AY*2+2:PRINT"  ":PLOT AX*32+144,400-AY*3
2:DRAWR 32,0:DRAWR 0,-32:DRAWR -32,0:DRAWR 0,32:RE
TURN
1030 FOR I=1 TO 5:PRINT CHR$(7);:FOR J=1 TO 30:NEX
T:NEXT:RETURN
1040 FOR I=16 TO 0 STEP-1:PRINT CHR$(7);:FOR J=1 T
O I*I+10:NEXT:NEXT:RETURN
1050 P=TEST(X*32+146,398-Y*32):RETURN
1060 T=0:IF XX>1 THEN X=XX-1:Y=YY:GOSUB 1050:T=T-(
P=0)
1070 IF XX<8 THEN X=XX+1:Y=YY:GOSUB 1050:T=T-(P=0)
1080 IF YY>1 THEN X=XX:Y=YY-1:GOSUB 1050:T=T-(P=0)
1090 IF YY<7 THEN X=XX:Y=YY+1:GOSUB 1050:T=T-(P=0)
1100 RETURN
```

PROGRAMME

Lignes 30 à 120 : initialisation

Après l'initialisation des nombres aléatoires (ligne 40), le programme saisit le niveau du jeu.

Lignes 200 à 260 : dessin du jeu

Les lignes 210 à 230 tracent le damier.

C'est la ligne 240 qui donne la position initiale du joueur et de l'ordinateur, les lignes 250 et 260 affichant les pions sur la grille.

Lignes 410 à 620 : coup du joueur

La première ligne vérifie si la partie est terminée. Si c'est le cas, on affiche le message « Perdu » et l'on recommence une partie dès qu'une touche est enfoncée.

Les lignes 430 à 490 déplacent le pion du joueur en fonction de la dernière touche appuyée (conservée dans D$). Le sous-programme en 1050 vérifie que la case choisie est libre.

Les lignes 510 à 620 bloquent la case choisie par le joueur.

Lignes 710 à 880 : coup de l'ordinateur

Comme pour le tour du joueur, la première ligne vérifie si la partie est terminée.

Les lignes 740 à 780 recherchent une case où le pion de l'ordinateur peut se déplacer. Pour que le jeu soit différent à chaque partie, on utilise des nombres aléatoires pour la sélection.

Les lignes 810 à 870 bloquent autant de cases que le niveau est élevé. Pour que la partie ne soit pas écourtée (surtout au niveau 3 !), on utilise aussi des nombres aléatoires pour le choix des cases à bloquer.

Lignes 1000 à 1100 : routines d'affichage

Ligne 1000 : efface le contenu de la case X, Y
Ligne 1010 : écrit un pion de couleur C dans la case X, Y
Ligne 1020 : efface le contenu de la case AX, AY
Ligne 1030 : émet un beep continu
Ligne 1040 : émet un beep discontinu
Ligne 1050 : place dans P la couleur de la case X, Y
Lignes 1060 à 1100 : placent dans T le nombre de déplacements possibles à partir de la case X, Y

Liste des variables :

AX et **AY**	ancienne position du joueur ou de l'ordinateur
C	couleur d'une case
CC	couleur initiale de la case X, Y
D	valeur ASCII de D$
D$	dernière touche appuyée
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
NIV	niveau du jeu
P	couleur de la case X, Y
T	nombre de déplacements à partir de X, Y
XJ et **YJ**	coordonnées du joueur
XO et **YO**	coordonnées de l'ordinateur
XX et **YY**	garde la position du joueur ou de l'ordinateur pour la recherche des cases libres
Z	indice de boucle FOR/NEXT



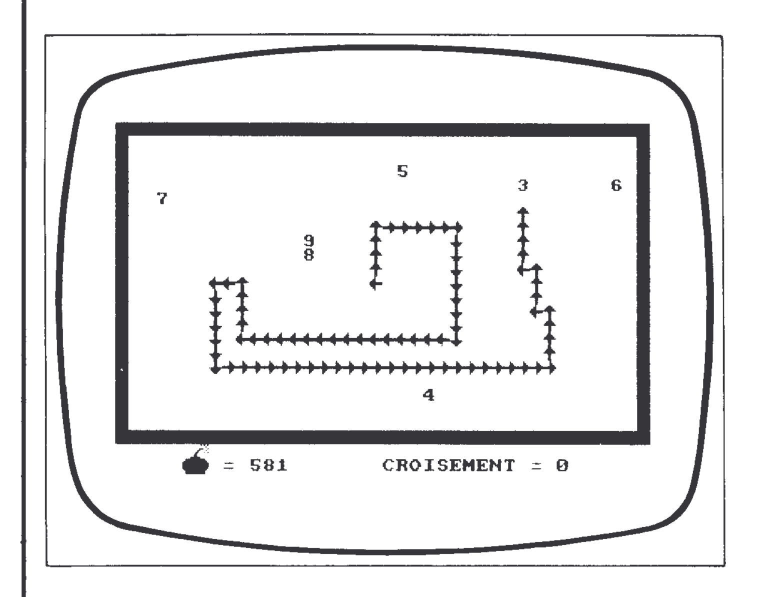
BOMBE A RETARDEMENT

Voilà un excellent jeu de réflexe explosif !

Vous avez exactement 1000 fractions de seconde pour relier entre eux dans un ordre croissant les 9 chiffres qui apparaisent dispersés sur l'écran. Pour cela, vous disposez des 4 flèches de déplacement.

Mais attention, un seul croisement est autorisé. Au deuxième, la bombe explosera.

C'est aussi ce qui risque d'arriver si vous ne respectez pas le temps imparti ou l'ordre des chiffres !



```
10 REM BOMBE A RETARDEMENT
20 REM ---------------------------------
30 SYMBOL AFTER 128:GOSUB 1000
40 RANDOMIZE TIME
50 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1
60 PAPER 1
70 FOR I=1 TO 40
80 LOCATE I,1:PRINT " ":LOCATE I,23:PRINT " "
90 IF I<24 THEN LOCATE 1,I:PRINT " ":LOCATE 40,I:P
RINT " "
100 NEXT I
110 PAPER 0:LOCATE 20,12:PRINT CHR$(129)
120 PEN 2
130 FOR I=1 TO 9
140 X(I)=INT(RND*36)+3:Y(I)=INT(RND*19)+3
150 IF TEST(X(I)*16-8,400-Y(I)*16)<>0 THEN 140
160 LOCATE X(I),Y(I):PRINT CHR$(I+48)
170 NEXT I
180 T=1000:P=1:C=0
190 PEN 3:LOCATE 6,24:PRINT CHR$(133)CHR$(134):LOC
ATE 6,25:PRINT CHR$(135)CHR$(136);" =";T;
200 LOCATE 21,25:PRINT "CROISEMENT = 0";
210 X=20:Y=12:LOCATE X,Y:PEN 1
280 REM ------------------------------
290 REM LE JEU
300 T=T-1:LOCATE 10,25:PRINT T;" ";:LOCATE X,Y
310 IF T=0 THEN 500
320 A$=INKEY$:IF A$="" THEN 300
330 IF A$=CHR$(242) AND  X>2 THEN V=-1:W=0
340 IF A$=CHR$(243) AND  X<39 THEN V=1:W=0
350 IF A$=CHR$(240) AND  Y>2 THEN V=0:W=-1
360 IF A$=CHR$(241) AND  Y<22 THEN V=0:W=1
370 IF V=0 AND W=0 THEN 300
380 IF TEST((X+V)*16-8,408-(Y+W)*16)=1 THEN C=C+1:
LOCATE 34,25:PRINT CHR$(48+C):IF C>1 THEN 500 ELSE
 420
390 A=TEST((X+V)*16-8,402-(Y+W)*16)
400 IF A=2 AND X(P)=X+V AND Y(P)=Y+W THEN P=P+1:IF
 P>9 THEN 700 ELSE 420
410 IF A<>0 THEN 500
420 X=X+V:Y=Y+W:LOCATE X,Y:PRINT CHR$(130+V+W*2)
430 V=0:W=0
440 GOTO 300
480 REM ------------------------------
490 REM PERDU
500 CLS:INK 1,0
510 FOR J=1 TO 2
520 FOR I=300 TO 120 STEP -20
530 PLOT I,500-I,J:DRAW 620-I,500-I,J:DRAW 620-I,I
-120,J:DRAW I,I-120,J:DRAW I,500-I,J
```

```
540 BORDER I MOD 27:SOUND 1,I,2
550 NEXT I
560 INK 2,1
570 NEXT J
580 RUN
680 REM ------------------------------
690 REM GAGNE
700 FOR I=700 TO 10 STEP-3:SOUND 1,I,2:NEXT
710 RUN
980 REM ------------------------------
990 REM GRAPHIQUES
1000 SYMBOL 128,24,60,126,255,24,24,24,24
1010 SYMBOL 129,16,48,112,255,255,112,48,16
1020 SYMBOL 131,8,12,14,255,255,14,12,8
1030 SYMBOL 132,24,24,24,24,255,126,60,24
1040 SYMBOL 133,0,0,0,0,0,1,1,7
1050 SYMBOL 134,21,2,57,98,197,128,128,224
1060 SYMBOL 135,63,127,255,255,255,255,127,63
1070 SYMBOL 136,252,254,255,255,255,255,254,252
1080 RETURN
```

PROGRAMME

Lignes 30 à 210 : initialisation

Après la définition des caractères graphiques par le sous-programme en 1000, on affiche le jeu en plaçant à des positions aléatoires les neuf chiffres à relier.

Lignes 300 à 440 : le jeu

La ligne 300 décrémente et affiche le temps qu'il reste avant l'explosion de la bombe. Si cette valeur est à zéro, on va directement en ligne 500 pour la fin du jeu.

Suivant la touche qui est appuyée, on modifie les incréments V et W.

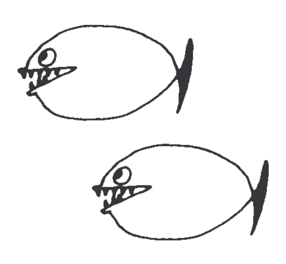
La ligne 380 teste le caractère que l'on vient de rencontrer. S'il s'agit d'une flèche, on incrémente le nombre de croisements C. Dès que celui-ci est supérieur à un, le programme exécute la phase explosion (lignes 500 à 580).

Dans le cas d'un chiffre, on incrémente le compteur P si l'ordre croissant a été respecté. Sinon on va aussi en ligne 500 pour la fin du jeu.

Avant de revenir en ligne 300, on affiche une flèche dans la direction du déplacement en utilisant les incréments V et W.

LISTE DES VARIABLES :

A$	dernière touche enfoncée
A	caractère sous le curseur
C	nombre de croisements
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
P	prochain chiffre à trouver
T	temps restant
V et **W**	incréments en X et Y
X et **Y**	coordonnées du joueur
X() et **Y()**	coordonnées des chiffres

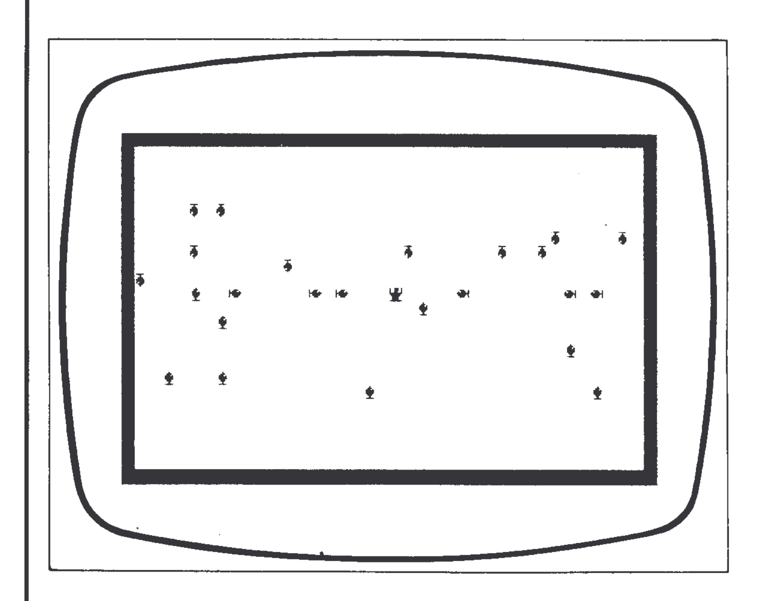


PIRANHAS

Ne paniquez pas si vous voulez sauver cet imprudent explorateur tombé dans l'Amazone au milieu des piranhas.

Sa seule chance est que ces féroces poissons s'entredévorent lorsqu'ils se retrouvent côte à côte.

Tout en déplaçant l'explorateur avec les quatre flèches, vous imaginerez sûrement une tactique pour forcer les piranhas à se rencontrer jusqu'au dernier. Vous devrez être prêt pour une autre traversée de l'Amazone, dès que vous aurez appuyé sur ENTER.



```
10 REM PIRANHAS
20 REM ---------------------------
30 SYMBOL AFTER 128:GOSUB 2000
40 DIM X(30),Y(30)
50 V(2)=-1:V(3)=1:W(1)=1:W(0)=-1
60 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
1:PEN 2:BORDER 1
70 FOR I=1 TO 40
80 LOCATE I,1:PRINT " ":LOCATE I,25:PRINT " ";
90 IF I<25 THEN LOCATE 1,I:PRINT " ":LOCATE 40,I:P
RINT " "
100 NEXT I:PAPER 0
110 RANDOMIZE TIME
120 P=INT(RND*11)+20
130 FOR I=1 TO P
140 X(I)=INT(RND*38)+2:Y(I)=INT(RND*22)+2
150 LOCATE X(I),Y(I):PRINT CHR$(128)
160 NEXT I
170 X=INT(RND*38)+2:Y=INT(RND*22)+2
180 PEN 3:LOCATE X,Y:PRINT CHR$(132)
190 IF INKEY$="" THEN 190
280 REM ---------------------------
290 REM DEPLACEMENT PIRANHAS
300 FOR I=1 TO P
310 IF X(I)=0 THEN 400
320 LOCATE X(I),Y(I):PRINT " "
330 FOR J=0 TO 3
340 A=TEST((X(I)+V(J))*16-8,408-(Y(I)+W(J))*16)
350 IF A=3 THEN 800
360 IF A=2 THEN 380
370 NEXT J:GOTO 410
380 X(I)=0:G=G+1:PRINT CHR$(7);
390 IF G=P-1 THEN 900
400 GOTO 600
410 IF Y(I)>Y THEN K=0:GOTO 450
420 IF Y(I)<Y THEN K=1:GOTO 450
430 IF X-1>=X(I) THEN K=3:GOTO 450
440 K=2
450 X(I)=X(I)+V(K):Y(I)=Y(I)+W(K)
580 REM ---------------------------
590 REM DEPLACEMENT JOUEUR
600 C$=INKEY$
610 IF C$="" THEN 700
620 FOR J=0 TO 3
630 IF C$=CHR$(240+J) THEN 650
640 NEXT J:GOTO 700
650 A=TEST((X+V(J))*16-8,408-(Y+W(J))*16):IF A=1 T
HEN 700
660 IF A=2 THEN 810
670 LOCATE X,Y:PRINT " "
680 X=X+V(J):Y=Y+W(J)
```

```
690 PEN 3:LOCATE X,Y:PRINT CHR$(132):D=D+1
700 IF X(I)>0 THEN PEN 2:LOCATE X(I),Y(I):PRINT CH
R$(128+K)
710 NEXT I
720 GOTO 300
780 REM -------------------------------
790 REM PERDU
800 LOCATE X+V(J),Y+W(J):PRINT " ":PEN 1:LOCATE X,
Y:PRINT CHR$(131):GOTO 820
810 LOCATE X,Y:PRINT " ":PEN 1:LOCATE X+V(J),Y+W(J
):PRINT CHR$(131)
820 FOR I=0 TO 26:PRINT CHR$(7);:BORDER I:FOR K=1
TO 40:NEXT:NEXT I
830 IF INKEY$<>CHR$(13) THEN 830
840 RUN
880 REM -------------------------------
890 REM GAGNE
900 PEN 1:LOCATE X,Y:PRINT CHR$(132)
910 FOR I=500 TO 50 STEP -5:SOUND 1,I,2:NEXT
920 LOCATE 13,25:PRINT D;"DEPLACEMENTS";
930 IF INKEY$<>CHR$(13) THEN 920
940 RUN
1980 REM ------------------------------
1990 REM GRAPHIQUES
2000 SYMBOL 130,0,57,93,255,125,57,0,0
2010 SYMBOL 131,0,156,186,255,190,156,0,0
2020 SYMBOL 129,124,16,56,124,124,92,56,16
2030 SYMBOL 128,8,28,58,62,62,28,8,62
2040 SYMBOL 132,157,157,153,255,126,60,60,255
2050 RETURN
```

PROGRAMME

Lignes 30 à 190 : initialisation

Après la définition des caractères graphiques par le sous-programme en 2000, on place dans les tableaux V() et W() les directions de déplacement des piranhas.

Le nombre de piranhas est déterminé par la variable P (nombre aléatoire). On remplit les tableaux X() et Y() avec les coordonnées des piranhas, X et Y donnant celles de l'explorateur.

Lignes 300 à 450 : le jeu

Chaque piranha est déplacé sur l'écran dans le sens qui le rapproche le plus de l'explorateur sauf s'il trouve un congénère sur une case adjacente. Dans ce cas, il s'empresse de l'avaler (ligne 360)

Lignes 600 à 720 : déplacement de l'explorateur

On déplace l'explorateur suivant la touche appuyée (saisie par INKEY$).

Lignes 800 à 940 : fin de partie

Suivant l'issue de la partie, Amstrad affiche un piranha (perdu!) ou un explorateur (gagné!).

LISTE DES VARIABLES :

A	couleur sous l'explorateur avant déplacement
C$	dernière touche enfoncée
G	nombre de piranhas éliminés
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
K	sens du déplacement des piranhas
P	nombre de piranhas
V() et **W()**	nombre de caractères de déplacement en X et en Y pour les 4 directions
X et **Y**	coordonnées de l'explorateur
X() et **Y()**	coordonnées des piranhas

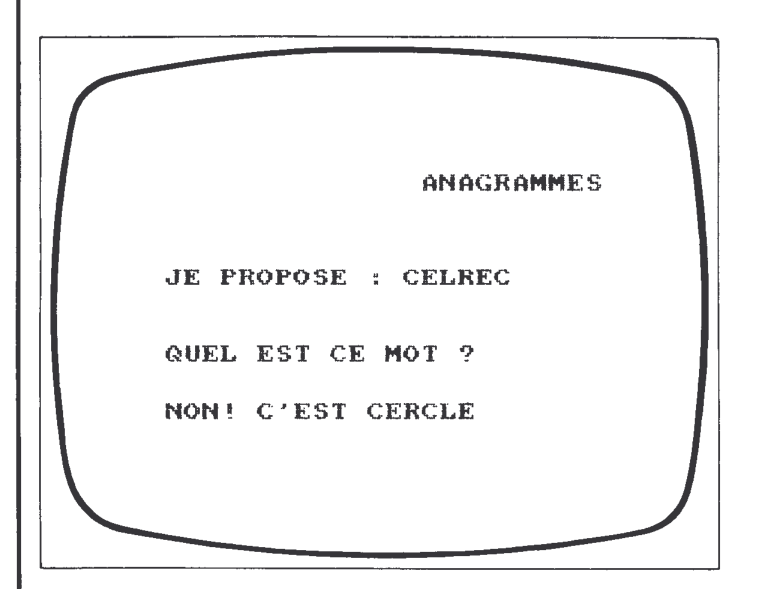


ANAGRAMMES

L'ordinateur choisit un mot parmi les 300 que contient le programme.

Puis, après un tour de passe-passe, il affiche un superbe anagramme.

A vous de trouver le mot d'origine. Si vous séchez lamentablement, taper uniquement sur ENTER pour obtenir la solution. En appuyant à nouveau sur ENTER, le programme vous propose un autre mot.



```
10 REM ANAGRAMMES
20 REM -------------------------------
30 MODE 1:INK 0,13:INK 1,0:PAPER 0:PEN 1:BORDER 13
40 DEFINT A-Z
50 DIM M$(300)
60 RANDOMIZE TIME
70 FOR I=1 TO 300
80 READ M$(I)
90 NEXT
180 REM ------------------------------
190 REM LE JEU
200 M=INT(RND*300)+1
210 Z$=M$(M)
220 FOR I=1 TO 10
230 A=INT(RND*6)+1:B=INT(RND*6)+1
240 X$=MID$(Z$,A,1)
250 MID$(Z$,A)=MID$(Z$,B,1)
260 MID$(Z$,B)=X$
270 NEXT
280 CLS:LOCATE 15,1:PRINT "ANAGRAMMES"
290 LOCATE 1,6:PRINT "JE PROPOSE : ";Z$
300 LOCATE 1,10:INPUT"QUEL EST CE MOT ";Z$
310 PRINT:PRINT
320 IF Z$=M$(M) THEN PRINT"OUI! C'EST ";Z$ ELSE PR
INT"NON! C'EST ";M$(M)
330 IF INKEY$<>CHR$(13) THEN 330
340 GOTO 200
480 REM ------------------------------
490 REM LES MOTS
500 DATA ABAQUE,ABBAYE,ABIMER,ABJECT,ABOYER,ABSENT
510 DATA ABSIDE,ABUSER,ACABIT,ACACIA,ACAJOU,ACCENT
520 DATA ACCORD,ACERBE,ACQUIT,ACTEUR,ACTION,ACTUEL
530 DATA ACUITE,ADAGIO,ADORER,ADULER,ADULTE,AERIEN
540 DATA AGACER,AGENCE,AGNEAU,AGREER,AIMANT,ALERTE
550 DATA ALOYAU,AMADOU,AMANDE,ANANAS,ANEMIE,ANGINE
560 DATA ANNEXE,ANODIN,APHONE,APLOMB,APPEAU,AUBURN
570 DATA BADAUD,BALISE,BALZAN,BAROUF,BASSET,BATEAU
580 DATA BAVARD,BECANE,BECTER,BELIER,BERGER,BETAIL
590 DATA BIELLE,BIGLER,BLAGUE,BLEUET,BOHEME,BOUGIE
600 DATA CABRER,CAGEOT,CALICE,CAMPER,CAPTER,CARDAN
610 DATA CARMIN,CARTON,CASINO,CAVEAU,CENTON,CERCLE
620 DATA CHACUN,CHAPON,CHASTE,CHEQUE,CHOMER,CISEAU
630 DATA CLASSE,CLIENT,CLIQUE,COCHER,COMBLE,CONDOR
640 DATA CONGRE,CONTRE,COPAIN,CORNET,CURARE,CURSIF
650 DATA DANGER,DEBILE,DECHET,DECOTE,DEDALE,DEESSE
660 DATA DELICE,DEMAIN,DEPART,DEPUIS,DESERT,DEVIER
670 DATA DEVOIR,DICTON,DINGUE,DIRECT,DIURNE,DRAGUE
680 DATA ECORCE,ECRIRE,EDITER,ELEVER,EMPIRE,ENCLIN
690 DATA ENIGME,ESCALE,ESCROC,ESPRIT,ETABLE,EXPRES
700 DATA FACHER,FARINE,FAUCON,FELURE,FESTIN,FIGURE
710 DATA FLACON,FLAQUE,FLEUVE,FORMAT,FOUINE,FRAISE
```

```
720 DATA FRAUDE,FRELON,FRIAND,FRISER,FUREUR,FURTIF
730 DATA GALANT,GATEAU,GAUFRE,GLAPIR,GOUJON,GOURDE
740 DATA HANCHE,HARGNE,HAUSSE,HARPIE,HUMIDE,HUMOUR
750 DATA IGNARE,ILLICO,IMPACT,INDIGO,INEPTE,INERTE
760 DATA INFAME,INFINI,INTACT,INTIME,IRONIE,IVOIRE
770 DATA JAMAIS,JAMBON,JOCKEY,JONQUE,JOVIAL,JUJUBE
780 DATA KARATE,KERMES,KIMONO,KIRSCH,KLAXON,KOPECK
790 DATA LABEUR,LABOUR,LACUNE,LANGUE,LIASSE,LICITE
800 DATA LIEVRE,LIMITE,LITIGE,LIVIDE,LOUCHE,LUSTRE
810 DATA MACHER,MADERE,MAFFIA,MALICE,MANDAT,MANGUE
820 DATA MANIOC,MARCHE,MARRON,MASQUE,MEDIUM,MEDUSE
830 DATA MEMBRE,MERLIN,MIETTE,MIOCHE,MOTEUR,MOUTON
840 DATA NARINE,NECTAR,NOMBRE,NOTICE,NOUGAT,NUANCE
850 DATA OCCASE,OCTAVE,OFFICE,ORACLE,OTARIE,OUVRIR
860 DATA PAELLA,PAPAYE,PAPIER,PAREIL,PARLER,PATATE
870 DATA PATRIE,PATRON,PAUMER,PAYSAN,PEIGNE,PELOTE
880 DATA PEPITE,PERCHE,PESTER,PIAULE,PILULE,PLANTE
890 DATA POIGNE,POISON,POPOTE,POUPEE,PRETER,PUZZLE
900 DATA QUARTZ,QUATRE,QUICHE,QUILLE,QUINTE,QUITTE
910 DATA RACINE,RADEAU,RAGOUT,RAISIN,RAMURE,RAPHIA
920 DATA RAVIER,REGARD,RELIER,RENDRE,RIFIFI,RUSTRE
930 DATA SAFRAN,SALAMI,SAPEUR,SAVOIR,SCRIPT,SEIGLE
940 DATA SHERIF,SINGER,SIRENE,SOLEIL,SOUPIR,SUIVRE
950 DATA TANDEM,TAPAGE,TAQUIN,TENACE,TESSON,TIMBRE
960 DATA TOPAZE,TOUPIE,TRAHIR,TRICOT,TRIPLE,TULIPE
970 DATA ULCERE,ULTIME,UNIQUE,URBAIN,URGENT,UTOPIE
980 DATA VALIDE,VANTER,VESTON,VIVIER,VOISIN,VOLUME
990 DATA WAPITI,WHISKY,YAOURT,ZENITH,ZIGZAG,ZOUAVE
```

Programme

Lignes 30 à 90 : initialisation

Après l'initialisation des nombres aléatoires, le programme place dans le tableau M$ la liste des 300 mots contenus dans les DATA des lignes 500 à 990.

Lignes 200 à 340 : le jeu

Le contenu aléatoire de la variable M pointe le mot que le joueur doit trouver.

Après avoir été placé dans la variable Z$, le mot est mélangé par la boucle FOR/NEXT des lignes 220 à 270.

Quand l'anagramme est affiché sur l'écran, le programme saisit la solution du joueur.

Si le résultat est erroné, la ligne 320 affiche le mot d'origine.

Lignes 500 à 990 : les mots

Si vous avez déjà tapé le programme MORSE, il est possible de récupérer les DATA en supprimant les lignes 10 à 410.

Liste des variables :

A et **B**	utilisées pour le mélange
I	indice de boucle FOR/NEXT
M	numéro du mot choisi
M$( )	liste des mots
X$	utilisée pour le mélange
Z$	mot proposé par le joueur ou l'ordinateur



AWARI

AWARI est un combat stratégique entre l'ordinateur et vous.

Amstrad vous demande d'entrer votre prénom ; puis, il dessine le jeu sur l'écran de cette façon :

	3	3	3	3	3	3	
0	PARTIE 1						0
	3	3	3	3	3	3	

EXEMPLE 1 :

Tout au cours d'une partie, l'ordinateur affichera votre prénom, puis un message, soit :

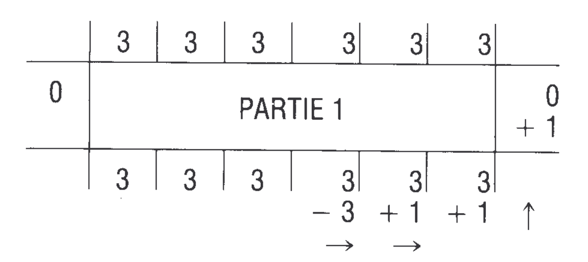
PRENOM
JE JOUE EN

ou :

PRENOM
QUEL EST VOTRE MOUVEMENT ?

L'ordinateur joue sur les cases du haut et le joueur sur les cases du bas.

Il y a trois pions au départ dans chaque case. Le jeu consiste à déplacer les trois pions d'une case de un en un dans les cases suivantes et dans le sens inverse des aiguilles d'une montre :



EXEMPLE 2 : MOUVEMENT

	3	3	3	3	3	3	
0	PARTIE 1						1
	3	3	3	0	4	4	

EXEMPLE 3 : RESULTAT

Il conviendra de considérer les cases sous les numéros suivants :



	6	5	4	3	2	1	
	3	3	3	3	3	3	
total de 0 l'ordin.							0 total du joueur
	3	3	3	3	3	3	
	1	2	3	4	5	6	

→
sens du jeu

EXEMPLE 4 :

Ainsi, dans l'exemple 2, la case mouvementée était la case 4 :

PRENOM
QUEL EST VOTRE MOUVEMENT ?
Taper 4.

Si le dernier pion placé s'arrête dans une case TOTAL, le joueur doit rejouer. Si le dernier pion s'arrête avant ou après cette case, c'est au joueur adverse de jouer. Dans l'exemple 2, c'est au joueur (et non à l'ordinateur) de rejouer.

La partie s'arrête quand toutes les cases de l'un des deux joueurs sont vides.

L'ordinateur signale alors qui a gagné. Le nombre de parties gagnées reste d'ailleurs affiché en permanence sur l'écran.

Puis le jeu initial se redessine, vous invitant à refaire une partie...

```
10 REM AWARI
20 REM ------------------------------
30 DIM B(14),K(14),G(14),H(14)
40 MODE 1:INK 0,1:INK 1,26:PAPER 0:PEN 1:BORDER 1
50 INPUT"VOTRE PRENOM";P$
60 PA=1
70 E=INT(RND+0.5)
80 CLS:PRINT
90 PRINT TAB(16);"AWARI"
100 LOCATE 1,18:PRINT P$
180 REM ----------------------------
190 REM AFFICHE LE JEU
200 FOR I=1 TO 13:B(I)=3:NEXT I
210 B(7)=0:B(14)=0
220 PLOT 0,200:DRAW 608,200
230 PLOT 0,232:DRAW 608,232
240 FOR I=66 TO 546 STEP 80
250 PLOT I,282:DRAW I,232:PLOT I,200:DRAW I,168
260 IF I=66 OR I=546 THEN PLOT I,232:DRAW I,200
270 NEXT I
280 LOCATE 1,12:PRINT B(14):LOCATE 15,12:PRINT"PAR
TIE:";PA:LOCATE 36,12:PRINT B(7)
290 FOR J=13 TO 8 STEP -1
300 LOCATE 1+(14-J)*5,10:PRINT B(J);" "
310 NEXT J
320 FOR J=1 TO 6
330 LOCATE 1+J*5,14:PRINT B(J);" "
340 NEXT J
380 REM ----------------------------
390 REM LE JEU
400 B=0
410 FOR I=1 TO 6
420 IF B(I)<>0 THEN B=1
430 NEXT I:IF B=0 THEN 1700
440 B=0
450 FOR I=8 TO 13
460 IF B(I)<>0 THEN B=1
470 NEXT I:IF B=0 THEN 1700
480 IF E=1 THEN 900
490 LOCATE 1,19:PRINT CHR$(18);"QUEL EST VOTRE MOU
VEMENT ? ";
500 A$=INKEY$:IF A$="" THEN 500 ELSE A=VAL(A$)
510 IF A<1 OR A>6 THEN 500
520 IF B(A)=0 THEN 500
530 LOCATE 28,19:PRINT CHR$(A+48):GOSUB 740
540 IF E=1 AND A=14 THEN 280
550 IF A=7 THEN 280
560 IF E=1 THEN E=0:GOTO 280
570 E=1:GOTO 280
680 REM ----------------------------
690 REM DEPLACEMENT DES PIONS
```

```
700 A=J
710 FOR Z=1 TO 14
720 B(Z)=G(Z)
730 NEXT Z
740 P=B(A):B(A)=0
750 FOR I=1 TO P
760 A=A+1:IF A=15 THEN A=1
770 B(A)=B(A)+1
780 NEXT I
790 RETURN
880 REM ------------------------------
890 REM COUP DE L'AMSTRAD
900 LOCATE 1,19:PRINT CHR$(18);"JE JOUE EN"
910 FOR J=1 TO 14
920 G(J)=B(J)
930 NEXT J
940 IF B(13)=1 THEN 1500
950 IF B(13)>=3 THEN 1500
960 IF B(12)>=3 THEN 1510
970 IF B(11)>=4 THEN 1520
980 FOR J=13 TO 8 STEP -1
990 IF G(J)=0 THEN 1070
1000 GOSUB 700
1010 IF A=14 THEN 1550
1020 H(J)=A
1030 FOR U=9 TO 13
1040 Z=U+B(U)
1050 IF Z=14 THEN K(J)=K(J)+1
1060 NEXT U
1070 NEXT J
1080 Z=0
1090 IF G(12)>=5 THEN 1510
1100 IF G(8)=1 THEN 1540
1110 IF G(9)=1 THEN 1530
1120 FOR J=8 TO 13
1130 U=K(J)
1140 IF U>Z THEN GOSUB 1560
1150 NEXT J
1160 IF Z>0 THEN 1280
1170 Z=8
1180 FOR J=8 TO 12
1190 U=H(J)
1200 IF U>Z AND U>0 THEN GOSUB 1560
1210 NEXT J
1220 IF A>7 THEN 1280
1230 Z=7
1240 FOR J=8 TO 13
1250 U=H(J)
1260 IF U<Z AND U>0 THEN GOSUB 1560
1270 NEXT J
1280 FOR U=8 TO 13
```

```
1290 K(U)=0
1300 H(U)=0
1310 NEXT U
1320 C=A-7
1330 LOCATE 12,19:PRINT CHR$(C+48)
1340 J=A
1350 GOSUB 700
1360 GOTO 540
1480 REM ------------------------------
1490 REM COUP CHOISI PAR L'AMSTRAD
1500 A=13:GOTO 1280
1510 A=12:GOTO 1280
1520 A=11:GOTO 1280
1530 A=9:GOTO 1280
1540 A=8:GOTO 1280
1550 A=J:GOTO 1280
1560 Z=U:A=J
1570 RETURN
1680 REM ------------------------------
1690 REM RESULTATS
1700 IF B(14)>B(7) THEN 1790
1710 IF B(14)<B(7) THEN 1750
1720 LOCATE 1,19:PRINT CHR$(18);"NOUS SOMMES EXAEQ
UO"
1730 E=INT(RND+0.5)
1740 GOTO 1820
1750 LOCATE 1,19:PRINT CHR$(18);"VOUS AVEZ GAGNE"
1760 T1=T1+1
1770 E=1
1780 GOTO 1820
1790 LOCATE 1,19:PRINT CHR$(18);"J'AI GAGNE"
1800 T2=T2+1
1810 E=0
1820 PA=PA+1
1830 LOCATE 3,22:PRINT "AMSTRAD:";T2;TAB(21);P$;":
";T1
1840 IF INKEY$<>CHR$(13) THEN 1840
1850 GOTO 200
```

Programme

Lignes 30 à 340 : initialisation et dessin du jeu

Après la déclaration des différents tableaux, le programme saisit le prénom du joueur et affiche à l'écran le dessin du jeu.

La variable E détermine le tour du joueur si sa valeur est 0, et le tour d'Amstrad si sa valeur est 1. Au premier tour, cette variable contient un nombre aléatoire,mais dès la deuxième partie, c'est toujours le perdant qui commence.

Lignes 400 à 480 : test de fin de partie

Le jeu doit s'arrêter quand un des joueurs n'a plus de pions dans son terrain. Dans ces lignes, on vérifie à chaque tour cette éventualité. Si c'est le cas, on passe à la ligne 1700.

En ligne 480, le programme se sépare en deux, suivant la valeur de E.

Lignes 490 à 570 : E=0, tour du joueur

Dès l'entrée du mouvement, on vérifie sa validité. En cas de mauvaise réponse, comme il n'est pas possible de passer son tour, la question est de nouveau posée jusqu'à l'obtention d'une réponse correcte.

L'appel du sous-programme de la ligne 740 déplace les pions des cases concernées.

Lignes 900 à 1570 : E=1, tour d'Amstrad

Pour déterminer son choix, Amstrad utilise trois tableaux pour comparer les solutions possibles et les plus avantageuses.

La solution trouvée est placée dans A comme pour le joueur, ce qui permet d'utiliser le sous-programme de déplacement des pions des lignes 700 à 790.

Lignes 700 à 790 : déplacement des pions

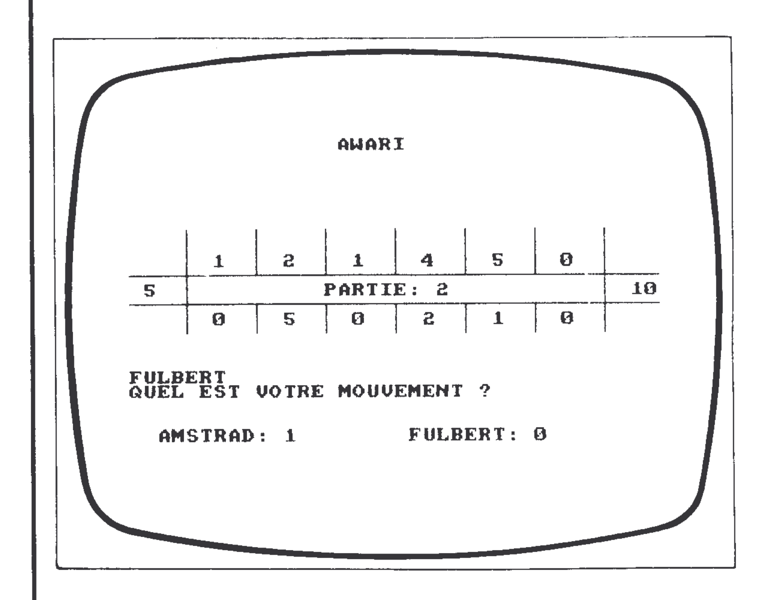
On déplace les pions de la case pointée par A en les distribuant dans les cases suivantes, en considérant que la case 15 (inexistante) correspond à la case 1.

Lignes 1700 à 1850 : résultats

Dès que l'un des joueurs n'a plus de points, on affiche à l'écran le nombre de parties gagnées par chacun ainsi qu'un message de félicitations.

Liste des variables :

A	votre mouvement (valeur décimale)
A$	dernière touche enfoncée
B	si B=0, fin de partie
B()	nombre de pions dans chaque case
C	numéro de la case jouée par Amstrad
E	si E=0 tour du joueur, si E=1 tour d'Amstrad
G()	sauvegarde du tableau B()
H()	nombre de pions dans chaque case après déplacement
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
K()	coefficient de chaque case
P	nombre de pions à déplacer
P$	votre prénom
PA	numéro de la partie
T1	nombre de parties gagnées par le joueur
T2	nombre de parties gagnées par Amstrad
U	coefficient de la case étudiée
Z	indice de boucle FOR/NEXT





FUSEE LUNAIRE

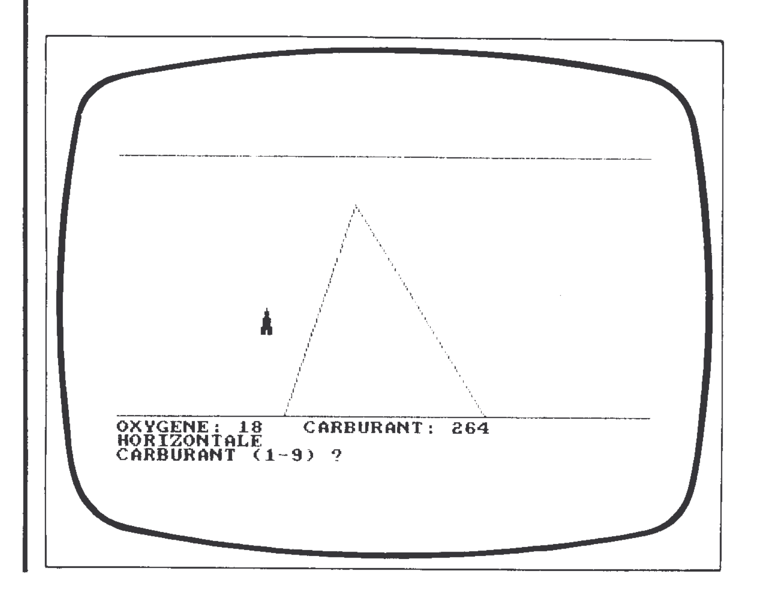
Un navigateur spatial s'entraîne en conduisant une fusée d'un côté à l'autre d'une montagne lunaire. Au début du jeu, la montagne se dessine sur l'écran, ainsi que la fusée et l'endroit opposé où elle doit retomber ; sont aussi indiquées les quantités d'oxygène et de carburant.

Le joueur doit donner la hauteur et la largeur du déplacement en deux temps chaque fois. Il doit :
- préciser le sens positif ou négatif,
- donner l'amplitude de 1 à 9,

L'oxygène et le carburant diminuent à chaque coup.

En cas d'erreur de navigation 6 messages peuvent apparaître :
- ECRASEE SUR LA MONTAGNE,
- SORTIE DE L'ESPACE,
- MISE EN ORBITE,
- LA FUSEE S'ECRASE,
- PLUS DE CARBURANT,
- PLUS D'OXYGENE.

Des paramètres précis régissent les déplacements de la fusée en fonction des données entrées par le joueur. Si vous ne parvenez jamais à poser la fusée, vous pouvez toujours jeter un coup d'œil au programme...



```
10 REM FUSEE LUNAIRE
20 REM ------------------------------
30 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1
40 SYMBOL AFTER 128
50 RANDOMIZE TIME
60 SYMBOL 128,8,8,28,28,28,28,62,62
70 SYMBOL 129,62,62,62,127,127,119,119,119
80 PLOT 0,399:DRAW 639,399:PLOT 0,100:DRAW 639,100
90 PLOT 540,100,2:DRAW 580,100
100 XM=INT(RND*120)+200
110 YM=350-INT(RND*50)
120 PLOT 200,100,1:DRAW XM,YM
130 DRAW 440,100
140 OX=20:CA=300
150 X=80:Y=134
160 TAG:MOVE X,Y:PRINT CHR$(128);:MOVE X,Y-16:PRIN
T CHR$(129);
280 REM ------------------------------
290 REM LE JEU
300 M$="VERTICALE":GOSUB 700:PV=Q
310 M$="HORIZONTALE":GOSUB 700:PH=Q
320 DX=X:DY=Y:AX=X:AY=Y
330 X=X+(PH+AH)*2:Y=Y+(PV+AV)*2
340 AH=PH/2:AV=PV/2
350 CA=CA-ABS(PH/2)-ABS(PV/2)
360 IF Y-DY=0 THEN INC=X-DX ELSE INC=(X-DX)/(Y-DY)
370 IF DX<X AND INC<0 THEN INC=-INC
380 IF DX>X AND INC>0 THEN INC=-INC
390 X=DX-INC:IV=1
400 IF DY>Y THEN IV=-1
410 FOR I=DY TO Y STEP IV
420 Y=I:X=X+INC:SOUND 1,600,1
430 IF Y>398 THEN 900
440 IF X>540 AND X<580 AND Y>132 AND Y<136 THEN 11
00
450 IF Y<134 THEN 910
460 IF X<5 OR X>635 THEN 920
470 TAG:MOVE AX,AY:PRINT" ";:MOVE AX,AY-16:PRINT"
";:MOVE X,Y:PRINT CHR$(128);:MOVE X,Y-16:PRINT CHR
$(129);:AX=X:AY=Y
480 IF TEST(X,Y-30)<>0 OR TEST(X+2,Y-32)<>0 THEN 9
30
490 IF TEST(X+8,Y-32)<>0 OR TEST(X+10,Y-32)<>0 THE
N 930
500 IF TEST(X+16,Y-30)<>0 OR TEST(X+14,Y-32)<>0 TH
EN 930
510 NEXT I
520 OX=OX-1
530 TAGOFF:LOCATE 1,20:PRINT CHR$(18);"OXYGENE:";O
X;"  CARBURANT:";CA
```

```
540 IF OX<=0 THEN 940
550 IF CA<=0 THEN 950
560 IF X>540 AND X<580 AND Y=134 THEN 1100
570 GOTO 300
680 REM ------------------------------
690 REM SAISIE
700 TAGOFF:PEN 3:LOCATE 1,21:PRINT CHR$(18);M$
710 LOCATE 1,22:PRINT CHR$(18);"SENS + OU - ?"
720 A$=INKEY$:IF A$="" THEN 720 ELSE IF A$=CHR$(13
) THEN A$="+"
730 IF A$<>"-" AND A$<>"+" THEN 720
740 LOCATE 1,22:PRINT CHR$(18);"CARBURANT (1-9) ?"
750 Q=VAL(INKEY$):IF Q<1 OR Q>9 THEN 750
760 Q=Q*2:IF A$="-" THEN Q=-Q
770 LOCATE 1,21:PRINT CHR$(18):PRINT CHR$(18)
780 PEN 1:RETURN
880 REM ------------------------------
890 REM PERDU
900 M$="MISE EN ORBITE":GOTO 960
910 M$="LA FUSEE S'ECRASE":GOTO 960
920 M$="SORTIE DE L'ESPACE":GOTO 960
930 M$="ECRASEE SUR LA MONTAGNE":GOTO 960
940 M$="PLUS D'OXYGENE":GOTO 960
950 M$="PLUS DE CARBURANT":GOTO 960
960 TAGOFF:PEN 3:LOCATE 1,21:PRINT M$:FOR I=50 TO
500 STEP 5:SOUND 1,I,2:NEXT
970 IF INKEY$<>CHR$(13) THEN 970
980 RUN
1080 REM ------------------------------
1090 REM GAGNE
1100 TAGOFF:LOCATE 1,21:PRINT"GAGNE":FOR I=500 TO
50 STEP-5:SOUND 1,I,2:NEXT
1110 IF INKEY$<>CHR$(13) THEN 1110
1120 RUN
```

Programme

Lignes 30 à 160 : initialisation

La ligne 50 initialise les nombres aléatoires.

Ce sont les lignes 80 à 130 qui dessinent la montagne. On définit la largeur et la hauteur de la montagne par deux nombres aléatoires XM et YM.

Lignes 300 à 570 : le jeu

Pour entrer le mouvement et la quantité de carburant nécessaire, on utilise le sous-programme en 700. On incrémente (ou l'on décrémente) la position de la fusée en fonction de ces valeurs. Les lignes 320 à 510 déplacent la fusée sur l'écran en prenant les anciennes coordonnées pour la position de départ et les coordonnées nouvellement calculées pour sa position d'arrivée.

Après affichage de l'oxygène et du carburant restant, on teste s'ils sont supérieurs à zéro. Si ce n'est pas le cas, on affiche un message correspondant au type d'erreur de vol commis (à partir de la ligne 900).

Dans le cas où les nouvelles coordonnées de la fusée correspondent à celles de l'arrivée, on saute aux lignes 1100 pour le message de félicitations.

Liste des variables :

AH et **AV**	accélération horizontale et verticale
A$	dernière touche appuyée
CA	carburant restant
DX et **DY**	sauvegarde de X et Y pendant le déplacement
I	indice de boucle FOR/NEXT
INC	incrément du déplacement horizontal
IV	incrément du déplacement vertical
M$	message à afficher
OX	oxygène restant
PH et **PV**	poussée horizontale et verticale
Q	quantité de carburant à utiliser
X et **Y**	coordonnées de la fusée
XM et **YM**	coordonnées de la montagne



CHASSE AU TRESOR

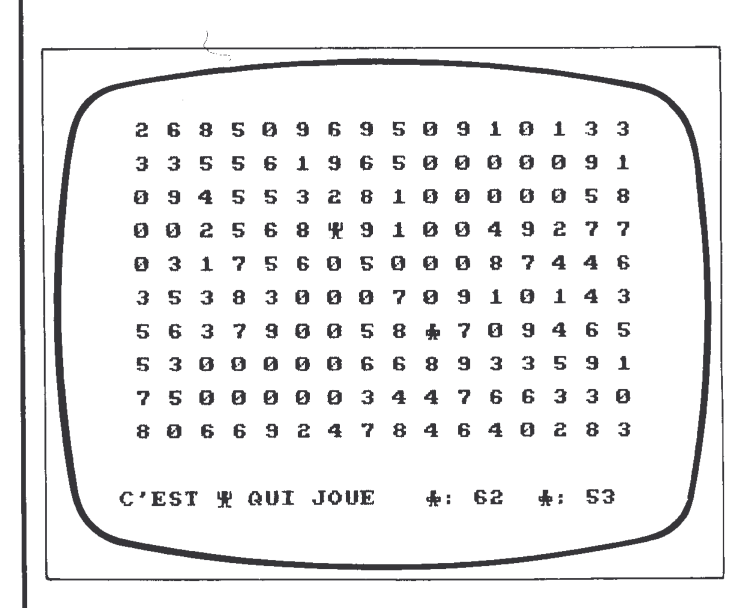
Vous n'avez pas de chance au jeu ? Alors voilà une chasse au trésor où la ruse et non le pur hasard vous sera utile pour gagner.

Cette chasse au trésor se joue à deux. Une grille de 16 sur 10 numéros s'inscrit devant vous. Il suffit de déplacer les deux chasseurs avec les quatre flèches du clavier selon l'itinéraire qui rapportera le plus de points (verticalement et horizontalement).

Attention, ne détruisez pas toute une zone sans prévoir un possible retour en arrière fructueux...

La partie s'arrête quand l'un des deux joueurs a obtenu au moins 250 points et les résultats s'affichent alors. Pour repartir, il faudra appuyer sur ENTER.

Maintenant bonne chasse !



```
10 REM CHASSE AU TRESOR
20 REM ------------------------------
30 SYMBOL AFTER 128
40 SYMBOL 128,93,93,89,127,28,28,20,119
50 SYMBOL 129,28,28,24,127,93,127,20,54
60 SYMBOL 130,93,93,89,127,28,28,20,119
70 DIM P(17,10)
80 X(1)=9:X(2)=10
90 Y(1)=6:Y(2)=5:C(1)=2:C(2)=3
100 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
 0:PEN 1:BORDER 1
110 RANDOMIZE TIME
120 FOR I=1 TO 10
130 FOR J=2 TO 17
140 P(J,I)=INT(RND*10):LOCATE J*2-1,I*2:PRINT P(J,
I)
150 NEXT J
160 NEXT I
170 LOCATE 3,24:PRINT "C'EST   QUI JOUE   ";:PEN 2
:PRINT CHR$(129);":      ";:PEN 3:PRINT CHR$(129);"
:"
280 REM ------------------------------
290 REM LE JEU
300 FOR I=1 TO 2
310 PEN 2:LOCATE X(1)*2,Y(1)*2:PRINT CHR$(127+I)
320 PEN 3:LOCATE X(2)*2,Y(2)*2:PRINT CHR$(128+I)
330 PEN C(I):LOCATE 9,24:PRINT CHR$(128)
340 AX=X(I):AY=Y(I)
350 A$=INKEY$:IF A$="" THEN 350
360 IF A$=CHR$(242) AND X(I)>2 THEN X(I)=X(I)-1:GO
TO 410
370 IF A$=CHR$(243) AND X(I)<17 THEN X(I)=X(I)+1:G
OTO 410
380 IF A$=CHR$(240) AND Y(I)>1 THEN Y(I)=Y(I)-1:GO
TO 410
390 IF A$=CHR$(241) AND Y(I)<10 THEN Y(I)=Y(I)+1:G
OTO 410
400 GOTO 350
410 B=P(X(I),Y(I)):P(X(I),Y(I))=0
420 PEN 1:LOCATE AX*2,AY*2:PRINT "0"
430 T(I)=T(I)+B
440 LOCATE 17+I*7,24:PRINT T(I)
450 IF T(I)>=250 THEN 600
460 NEXT I
470 GOTO 300
580 REM ------------------------------
590 REM RESULTATS
600 CLS
610 IF T(1)>T(2) THEN I=1:GOTO 630
620 I=2
630 PEN 2:LOCATE 15,11:PRINT CHR$(127+I);":";T(1)
640 PEN 3:LOCATE 15,13:PRINT CHR$(128+I);":";T(2)
650 PEN 1:LOCATE 9,15:PRINT"C'EST ";:PEN C(I):PRIN
T CHR$(128);:PEN 1:PRINT " QUI GAGNE"
660 IF INKEY$<>CHR$(13) THEN 660
670 RUN
```

PROGRAMME

Lignes 30 à 170 : initialisation

Les tableaux X() et Y() donnent les coordonnées de départ des deux joueurs.

On affiche à l'écran une grille de nombres aléatoires de 10 sur 16. Ces nombres sont sauvés dans le tableau P.

Lignes 300 à 470 : le jeu

On vérifie les valeurs données par le caractère saisi au clavier pour déplacer les joueurs.

Après déplacement des figures, on ajoute au total des points du joueur le nombre sur lequel on vient de se placer. On récupère cette valeur dans le tableau P. Ce total est affiché au bas de l'écran.

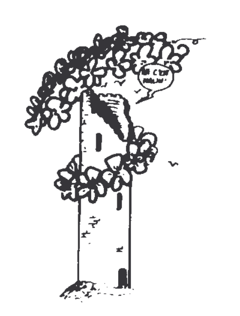
On vérifie après chaque tour si le joueur ne dépasse pas les 250 points requis pour gagner (tableau **T()**).

Lignes 600 à 670 : résultats

Un des joueurs ayant dépassé 250 points, on affiche à l'écran les résultats de la partie.

LISTE DES VARIABLES :

A$	déplacement
B	valeur sous le joueur
AX et **AY**	coordonnées avant déplacement
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
P()	tableau des points
T()	total des points de chaque joueur
X() et **Y()**	coordonnées des joueurs



TOURS DE HANOI

Face à ces tours de Hanoi, vous devrez faire preuve de stratégie, de réflexion et de beaucoup de patience. Il s'agit de faire passer sur la tour de droite et dans le même ordre un certain nombre de disques empilés sur la tour de gauche en vous aidant de la tour du milieu.

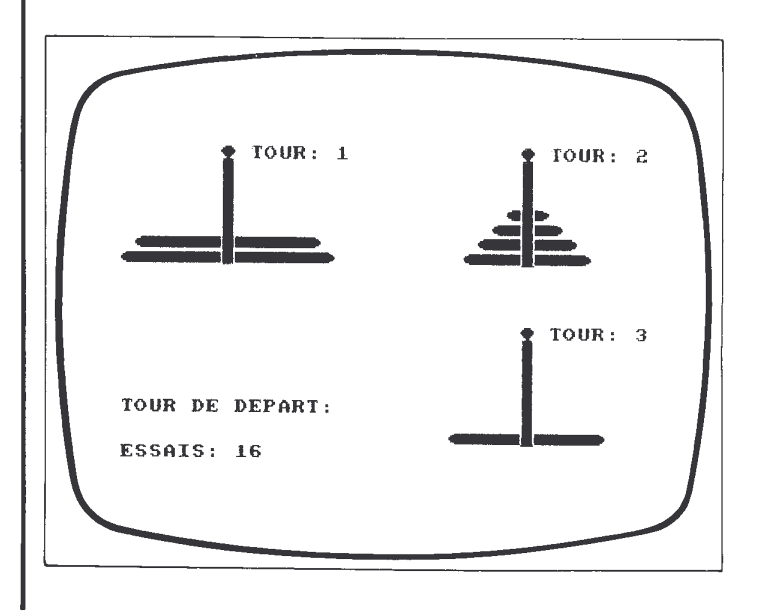
Vous pouvez choisir le nombre de disques à manœuvrer entre trois et sept.

Amstrad demande en bas de l'écran la tour de départ du mouvement, puis la tour d'arrivée : entrer 1, 2 ou 3, numéros des tours.

Il refuse les ordres impossibles à exécuter : on ne peut poser un disque sur un autre moins large. Par contre un disque peut aller de la tour 1 à la tour 3, puis revenir à la tour 2.

Il affiche aussi tout au long du jeu le nombre d'essais.

Petit conseil : il vaut mieux pour vos nerfs commencer par trois disques !



```
10 REM TOURS DE HANOI
20 REM ------------------------------
30 SYMBOL AFTER 128:GOSUB 2000
40 MODE 1:INK 0,1:INK 1,24:INK 2,15:INK 3,6:PAPER
0:PEN 1:BORDER 1
50 INPUT"NOMBRE DE DISQUES";N
60 IF N<3 OR N>7 THEN 40
70 CLS
80 DIM D(3,7)
90 X(1)=8:X(2)=29:X(3)=29
100 Y(1)=9:Y(2)=9:Y(3)=21
110 FOR I=1 TO 7
120 IF I<=N THEN D(1,I)=8-I
130 NEXT I
140 GOSUB 1500
150 E=0
180 REM ------------------------------
190 REM LE JEU
200 PEN 1:LOCATE 1,18:PRINT"TOUR DE DEPART:";
210 A$=INKEY$:IF A$="" THEN 210
220 TD=VAL(A$):IF TD<1 OR TD>3 THEN 210
230 J=TD:GOSUB 800:IF Q=1 THEN 200
240 DD=I
250 LOCATE 1,18:PRINT"TOUR D'ARRIVEE:";
260 A$=INKEY$:IF A$="" THEN 260
270 TA=VAL(A$):IF TA<1 OR TA>3 THEN 260
280 IF TA=TD THEN 200
290 J=TA:GOSUB 800:IF Q=1 THEN 310
300 IF D(TD,DD)>D(TA,I) THEN 200
310 DA=I+1
320 D(TA,DA)=D(TD,DD):D(TD,DD)=0
330 I=TD:J=DD:GOSUB 1000
340 I=TA:J=DA:GOSUB 1000
350 E=E+1
360 PEN 3:LOCATE 1,21:PRINT"ESSAIS:";:PEN 1:PRINT
E
370 FOR I=1 TO N
380 IF D(3,I)=0 THEN 200
390 NEXT I
400 PEN 2:LOCATE 1,18:PRINT"REUSSI EN";:PEN 1:PRIN
T E;:PEN 2:PRINT "COUPS"
410 IF INKEY$="" THEN 410
420 RUN
780 REM ------------------------------
790 REM NUMERO DU DISQUE SUPERIEUR
800 FOR I=7 TO 1 STEP -1
810 IF D(J,I)<>0 THEN Q=0:GOTO 830
820 NEXT I:Q=1
830 RETURN
980 REM ------------------------------
990 REM AFFICHE LES DISQUES
```

```
1000 PEN 2:FOR K=7 TO 1 STEP -1
1010 IF D(I,J)>K THEN LOCATE X(I)-K,Y(I)-J:PRINT C
HR$(132):GOTO 1040
1020 IF D(I,J)=K THEN LOCATE X(I)-K,Y(I)-J:PRINT C
HR$(131):GOTO 1040
1030 LOCATE X(I)-K,Y(I)-J:PRINT " "
1040 NEXT K
1050 FOR K=1 TO 7
1060 IF D(I,J)=K THEN LOCATE X(I)+K,Y(I)-J:PRINT C
HR$(130):GOTO 1090
1070 IF D(I,J)>K THEN LOCATE X(I)+K,Y(I)-J:PRINT C
HR$(132):GOTO 1090
1080 LOCATE X(I)+K,Y(I)-J:PRINT " "
1090 NEXT K
1100 RETURN
1480 REM ------------------------------
1490 REM DESSIN DU JEU
1500 FOR I=1 TO 3
1510 PEN 2:LOCATE X(I),Y(I)-8:PRINT CHR$(128);:PEN
 3:PRINT " TOUR:";:PEN 1:PRINT I:PEN 2
1520 FOR J=1 TO 7
1530 GOSUB 1000
1540 IF J>1 THEN LOCATE X(I),Y(I)-J:PRINT CHR$(129
):GOTO 1560
1550 LOCATE X(I),Y(I)-J:PRINT CHR$(133)
1560 NEXT J
1570 NEXT I
1580 RETURN
1980 REM ------------------------------
1990 REM GRAPHIQUES
2000 SYMBOL 128,60,126,255,255,126,60,24,126
2010 SYMBOL 129,126,126,126,126,126,126,126,126
2020 SYMBOL 130,0,248,254,255,255,254,248,0
2030 SYMBOL 131,0,31,127,255,255,127,31,0
2040 SYMBOL 132,0,255,255,255,255,255,255,0
2050 SYMBOL 133,126,126,126,126,126,126,126,255
2060 RETURN
```

PROGRAMME

Lignes 30 à 150 : initialisation

Après la définition des caractères graphiques par le sous-programme en 2000, on saisit le nombre de disques à jouer entre trois et sept.

On dimensionne un tableau D pour connaître pendant le jeu l'emplacement des sept disques sur les trois tours.

On remplit le tableau de départ en mettant tous les disques sur la première tour, le disque le plus grand étant placé à la position 1. Puis l'on fait appel au sous-programme en 1500 pour le dessin du jeu.

Lignes 200 à 360 : déplacement des disques

On commence le jeu en saisissant le numéro de la tour de départ (on déplace dans ce cas le disque supérieur). Le sous-programme en 800 trouve le numéro de ce disque. Si cette tour ne comporte aucun disque, on met le drapeau à 1 (variable Q). Au retour du sous-programme, si Q=1 on revient à la saisie de la tour de départ.

On utilise le même principe pour la tour d'arrivée. Au retour du sous-programme en 800, on teste si le disque du dessus n'est pas plus petit. Si c'est le cas, on revient à la saisie de la tour de départ.

Lignes 370 à 420 : fin du jeu

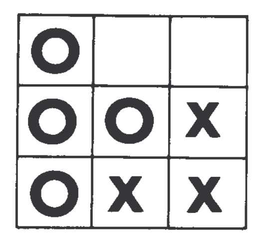
On teste sur la troisième tour si tous les disques sont déjà placés. Si ce n'est pas le cas, on retourne à la ligne 200 pour l'essai suivant, sinon on affiche le message REUSSI et le nombre d'essais.

Lignes 1000 à 1100 : dessin des disques

Ces lignes dessinent sur l'écran avec des caractères graphiques les disques tels qu'ils se trouvent dans le tableau D().

LISTE DES VARIABLES :

A$	dernière touche enfoncée
D()	emplacement des disques sur les tours
DA	emplacement du disque sur la tour d'arrivée
DD	emplacement du disque sur la tour de départ
E	nombre d'essais
I et **J**	coordonnées du disque à afficher
K	indice de boucle FOR/NEXT
N	nombre de disques
Q	si Q=1, il y a erreur de déplacement
TA	tour d'arrivée
TD	tour de départ
X() et **Y()**	coordonnées des trois tours



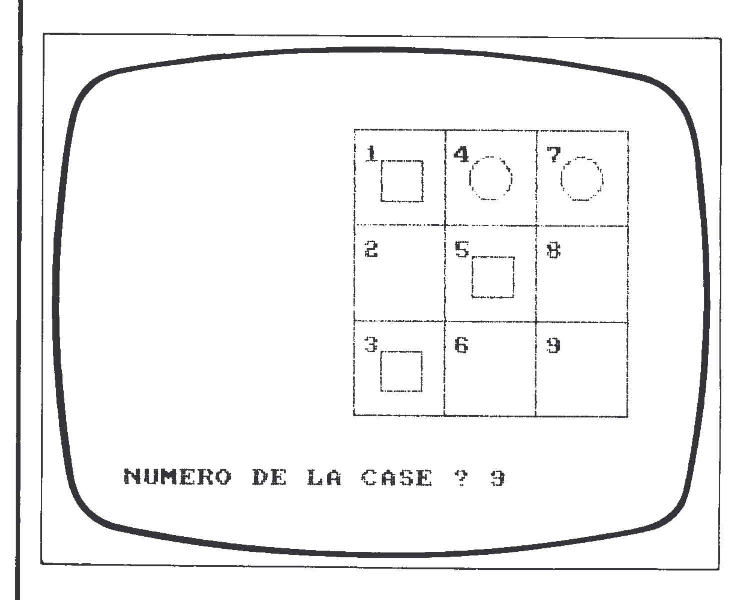
OXO

Prenez une grille de trois sur trois, deux joueurs et des pions différents pour chacun d'eux. Vous obtenez l'un des plus vieux jeux du monde que les hommes ont d'abord dessiné puis effacé des milliers de fois sur le sable.

Ici chaque joueur place un pion à tour de rôle en indiquant un numéro de case sur la grille. Chacun doit empêcher l'autre d'en aligner trois, sans quoi il perd.

La partie se joue très serrée contre l'ordinateur qui ne laissera passer aucune étourderie. Il suffit d'appuyer sur ENTER pour commencer une autre partie.

Méfiez-vous ! Votre Amstrad est très difficile à battre.



```
10 REM OXO
20 REM ---------------------------------
30 DIM I(4),J(4):DEG
40 T=0
50 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1
60 FOR J=1 TO 3
70 FOR I=0 TO 2
80 PLOT 200+I*80,460-J*80:DRAWR 80,0:DRAWR 0,-80:D
RAWR -80,0:DRAWR 0,80
90 PEN 2:LOCATE 14+I*5,J*5-2:PRINT RIGHT$(STR$(I*3
+J),1):PEN 1
100 NEXT:NEXT
110 GOTO 400
120 IF T=9 THEN 1140
130 PEN 3:LOCATE 1,20:PRINT CHR$(7);CHR$(18);"NUME
RO DE LA CASE ?";:PEN 1
140 A=VAL(INKEY$):IF A<1 OR A>9 THEN 140 ELSE PRIN
T A
150 X=INT(A/3.5):Y=A-(X*3)
160 IF P(X+1,Y)<>0 THEN 130
170 FOR I=1 TO 360 STEP 6
180 MOVE 240+X*80,420-Y*80:PLOTR 18*COS(I),18*SIN(
I):NEXT
190 P(X+1,Y)=1
200 T=T+1
380 REM ------------------------------
390 REM COUP DE L'ORDINATEUR
400 IF P(2,2)=0 THEN X=2:Y=2:GOTO 680
410 ERASE I,J:DIM I(4),J(4):P=1:GOSUB 800
420 FOR I=1 TO 4
430 IF I(I)=3 OR J(I)=3 THEN 1100
440 NEXT
450 ERASE I,J:DIM I(4),J(4):P=2:GOSUB 800
460 FOR I=1 TO 4
470 IF I(I)=2 OR J(I)=2 THEN 500
480 NEXT
490 P=1
500 ERASE I,J:DIM I(4),J(4)
510 X=0:Y=0:M=0
520 GOSUB 800
530 FOR I=1 TO 4
540 IF I(I)=3 OR J(I)=3 THEN 1100
550 IF I(I)>=M THEN X=I:Y=0:M=I(I)
560 IF J(I)>=M THEN Y=I:X=0:M=J(I)
570 NEXT
580 IF T=9 THEN 1140
590 IF P=1 AND M=2 THEN 610
600 P=P+1:IF P<3 THEN 500 ELSE P=2
610 IF X=4 THEN IF P(1,1)=0 THEN X=1:Y=1:GOTO 680
ELSE X=3:Y=3:GOTO 680
620 IF Y=4 THEN IF P(1,3)=0 THEN X=1:Y=3:GOTO 680
ELSE X=3:Y=1:GOTO 680
630 IF X=0 AND Y=0 THEN X=1
640 FOR I=1 TO 3
650 IF Y=0 AND P(X,I)=0 THEN Y=I
660 IF X=0 AND P(I,Y)=0 THEN X=I
670 NEXT
680 P(X,Y)=2
```

```
690 PEN 3:LOCATE 1,20:PRINT CHR$(7);CHR$(18);"JE J
OUE EN";:PEN 1:PRINT (X-1)*3+Y
700 PLOT 145+X*80,435-Y*80:DRAWR 35,0:DRAWR 0,-35:
DRAWR -35,0:DRAWR 0,35
710 FOR I=1 TO 2000:NEXT
720 IF P=2 AND M=2 THEN 1100
730 T=T+1:GOTO 120
780 REM ------------------------------
790 REM COMPTE LES CASES
800 C=0:D=0
810 FOR I=1 TO 3
820 A=0:B=0
830 FOR J=1 TO 3
840 IF P(I,J)=P THEN I(I)=I(I)+1
850 IF P(I,J)<>0 THEN A=A+1
860 IF P(J,I)=P THEN J(I)=J(I)+1
870 IF P(J,I)<>0 THEN B=B+1
880 NEXT J
890 IF A=3 AND I(I)<>3 THEN I(I)=0
900 IF B=3 AND J(I)<>3 THEN J(I)=0
910 IF P(I,I)=P THEN I(4)=I(4)+1
920 IF P(I,I)<>0 THEN C=C+1
930 IF P(I,4-I)=P THEN J(4)=J(4)+1
940 IF P(I,4-I)<>0 THEN D=D+1
950 NEXT I
960 IF C=3 AND I(4)<>3 THEN I(4)=0
970 IF D=3 AND J(4)<>3 THEN J(4)=0
980 RETURN
1080 REM ------------------------------
1090 REM RESULTATS
1100 LOCATE 1,20:PRINT CHR$(18);
1110 IF P=1 THEN PRINT"VOUS GAGNEZ"
1120 IF P=2 THEN PRINT"JE GAGNE"
1130 GOTO 1160
1140 LOCATE 1,20:PRINT CHR$(18);
1150 PRINT"PARTIE NULLE"
1160 FOR I=16 TO 0 STEP-1:PRINT CHR$(7);:FOR J=1 T
O I*I+10:NEXT:NEXT
1170 LINE INPUT A$:RUN
```

LISTE DES VARIABLES :

A	choix du joueur
A, B, C, D	nombre de pions déjà posés dans une ligne, colonne ou diagonale
A$	dernière touche appuyée
I	indice de boucle FOR/NEXT
I( )	nombre de pions dans chaque colonne
J	indice de boucle FOR/NEXT
J( )	nombre de pions dans chaque ligne
M	nombre de pions autour de la case choisie
P	P=1 pion du joueur, P=2 pion de l'ordinateur
P(,)	position des pions dans la grille
T	nombre de pions déjà posés
X et **Y**	coordonnées sur la grille du coup du joueur ou de l'ordinateur

Programme

Lignes 30 à 120 : initialisation

Les boucles FOR/NEXT des lignes 60 à 100 affichent le dessin du jeu.

Lignes 120 à 200 : coup du joueur

Après avoir testé si la partie est terminée (T=9), le programme saisit le choix du joueur. Les lignes 140 et 160 vérifient que le coup est possible, et les lignes 170 et 180 affichent le pion.

Lignes 400 à 730 : coup de l'ordinateur

Les lignes 400 à 440 vérifient si le joueur a gagné en utilisant le sous-programme en 800 qui compte le nombre de pions dans chaque ligne, colonne et diagonale.

Les lignes 450 à 490 utilisent le même principe mais pour les pions de l'ordinateur. Si deux pions sont déjà alignés, on ne cherche plus à bloquer l'adversaire.

Les lignes 500 à 570 recherchent la meilleure case à jouer, donc celle où il y a le plus de pions de l'adversaire ou de ses propres pions. La meilleure valeur est conservée dans la variable M.

Si l'adversaire a déjà deux pions d'alignés (ligne 590), on va directement en ligne 610 pour occuper la troisième case.

Les lignes 610 à 670 recherchent une solution si les lignes précédentes n'en avaient pas trouvée.

Ce sont les lignes 690 et 700 qui affichent le coup choisi par l'ordinateur.

Lignes 800 à 980 : compte les cases

Ce sous-programme compte le nombre de pions P de chaque ligne, colonne ou diagonale. Si trois pions (de n'importe quel joueur) sont déjà alignés, on ne tient pas compte des tableaux I et J (rôle des variables A, B, C et D).



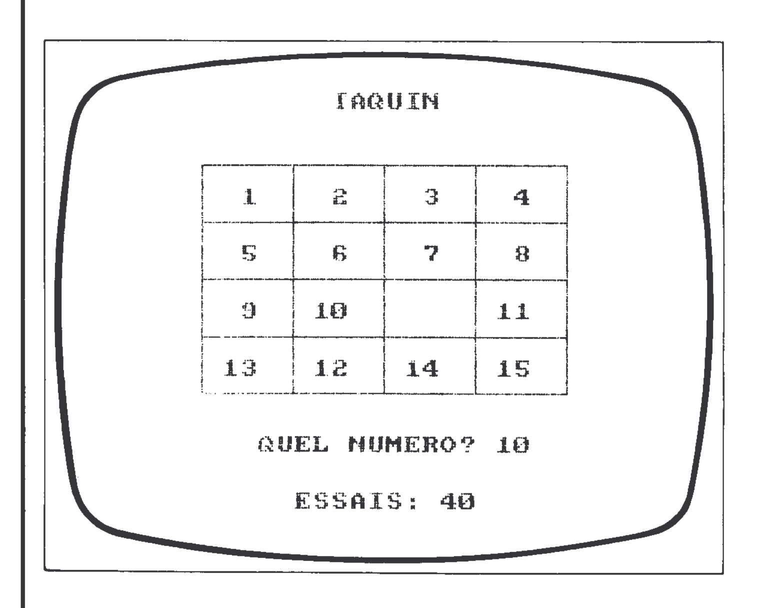
TAQUIN

Le taquin ou puzzle à 15 a été inventé par Sam LLOYD aux Etats-Unis dans les années 1880. Il se compose d'un plateau et de quinze plaquettes numérotées de 1 à 15 que l'on dispose en désordre (plus de vingt mille milliards de possibilités). Le vide sur le plateau permet de bouger les plaquettes pour les remettre dans leur ordre croissant.

Lancez le programme en tapant RUN. Après quelques instants, le temps de mélanger le taquin, le plateau s'affiche à l'écran.

Seules les plaquettes voisines de la case vide peuvent se déplacer.

Il ne s'agit pas du jeu original de Sam LLOYD. En conséquence, toutes les positions de départ permettent une reconstitution de l'ensemble du jeu.



```
10 REM TAQUIN
20 REM ------------------------------
30 MODE 1:INK 0,1:INK 1,24:INK 2,15:INK 3,6:PAPER
0:PEN 1:BORDER 1
40 DIM T(16)
50 D(1)=-4:D(2)=4:D(3)=1:D(4)=-1
60 PEN 3:PRINT TAB(17);"TAQUIN":PEN 1
70 RANDOMIZE TIME
180 REM -----------------------------
190 REM MELANGE DU TAQUIN
200 FOR I=1 TO 15:T(I)=I:NEXT I
210 V=16:E=0
220 FOR I=1 TO 128
230 A=INT(RND*4)+1
240 IF V+D(A)<1 OR V+D(A)>16 THEN 230
250 IF INT(V/4)=V/4 AND A=3 THEN 230
260 IF INT(V/4)=(V-1)/4 AND A=4 THEN 230
270 T(V)=T(V+D(A))
280 T(V+D(A))=0
290 V=V+D(A)
300 NEXT I
380 REM -----------------------------
390 REM DESSIN DU TAQUIN
400 FOR I=222 TO 462 STEP 80
410 PLOT 142,338:DRAW I,338:DRAW I,146:DRAW 142,14
6:DRAW 142,338
420 NEXT I
430 FOR I=110 TO 254 STEP 48
440 PLOT 142,338:DRAW 462,338:DRAW 462,400-I:DRAW
142,400-I:DRAW 142,338
450 NEXT I:PEN 2
460 FOR I=0 TO 15
470 IF T(I+1)=0 THEN 510
480 IF T(I+1)<10 THEN 500
490 LOCATE 10+5*(I-INT(I/4)*4),6+INT(I/4)*3:PRINT
T(I+1):GOTO 510
500 LOCATE 11+5*(I-INT(I/4)*4),6+INT(I/4)*3:PRINT
T(I+1)
510 NEXT I
580 REM -----------------------------
590 REM LE JEU
600 PEN 1:LOCATE 13,19:PRINT CHR$(18);:INPUT"QUEL
NUMERO";R$:R=VAL(R$)
610 IF R<1 OR R>15 THEN 600
620 FOR I=1 TO 16
630 IF R=T(I) THEN 650
640 NEXT I
650 R=I
660 FOR I=1 TO 4
670 IF R+D(I)>16 OR R+D(I)<1 THEN 690
680 IF T(R+D(I))=0 THEN 710
```

```
690 NEXT I
700 GOTO 600
710 T(R+D(I))=T(R)
720 T(R)=0
730 R=R-1
740 LOCATE 11+5*(R-INT(R/4)*4),6+INT(R/4)*3:PRINT"
  "
750 J=R+D(I):PEN 2
760 IF T(J+1)<10 THEN 780
770 LOCATE 10+5*(J-INT(J/4)*4),6+INT(J/4)*3:PRINT
T(J+1):GOTO 790
780 LOCATE 11+5*(J-INT(J/4)*4),6+INT(J/4)*3:PRINT
T(J+1)
790 E=E+1:PEN 3:LOCATE 15,22:PRINT"ESSAIS:";E
880 REM --------------------------------
890 REM FIN DU JEU
900 FOR I=1 TO 15
910 IF T(I)<>I THEN 600
920 NEXT I
930 LOCATE 10,19:PRINT"O.K VOUS AVEZ TROUVE"
940 IF INKEY$<>CHR$(13) THEN 940
950 RUN
```

LISTE DES VARIABLES :

A	nombre aléatoire pour le mélange du taquin
D()	direction du déplacement
E	nombre d'essais
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
R$	réponse du joueur
R	numéro de la case à déplacer
T()	contient les nombres du taquin
V	pointeur de la case vide

Programme

Lignes 30 à 300 : initialisation et mélange

Le tableau T correspond aux seize cases du taquin. On y place donc les numéros de 1 à 15 (ligne 200). Le tableau D correspond au déplacement des plaquettes.

Tableau T()

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	/

Tableau D()

```
        - 4
         ↑
- 1 ←    X    → + 1
         ↓
        + 4
```

On mélange les numéros en permutant deux cases désignées par un nombre aléatoire et le pointeur de la case vide (variable V). Cent vingt-huit permutations sont nécessaires pour mélanger correctement le taquin.

Lignes 400 à 510 : dessin du taquin

On affiche le jeu à l'écran dans le nouvel ordre (ou plutôt désordre !).

Lignes 600 à 790 : le jeu

On déplace le numéro tapé par le joueur en vérifiant qu'il n'est pas inférieur à 1 ou supérieur à 15, et qu'il est bien placé à côté de la case vide. Le déplacement est calculé avec le tableau D.

Lignes 900 à 950 : fin du jeu

A chaque essai on vérifie qu'il reste des plaquettes dans le désordre. Si c'est le cas on retourne à la ligne 600 pour un nouvel essai.



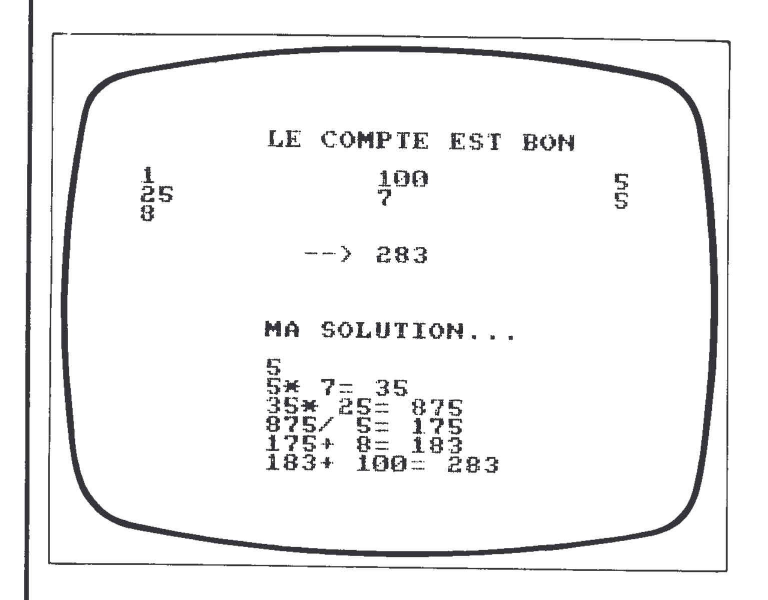
LE COMPTE EST BON

Il n'est peut-être pas encore 19 heures mais rien ne vous empêche de commencer dès maintenant à vous entraîner au « compte est bon ».

Au bout d'un certain temps de réflexion, Amstrad vous présente sept nombres aléatoires qu'il vous faudra combiner avec les quatre opérations pour trouver le huitième indiqué au milieu de l'écran.

Cinquante secondes plus tard, votre ordinateur affiche alors sa solution.

Il vous reste encore du temps ? Alors tapez sur le clavier (pas trop fort !) et une autre série de nombres apparaîtra.



```
10 REM LE COMPTE EST BON
20 REM ------------------------------
30 MODE 1:INK 0,1:INK 1,24:INK 2,6:PAPER 0:PEN 2:B
ORDER 1:PRINT TAB(11);" LE COMPTE EST BON":PRINT:P
EN 1
40 RANDOMIZE TIME:NC=INT(RND*4)+4
50 S$=""
60 FOR I=1 TO NC
70 N(I)=INT(RND*14)+1
80 IF N(I)<11 THEN 100
90 N(I)=(N(I)-10)*25
100 IF I>1 THEN 130
110 R=N(1):S$=STR$(N(1))
120 GOTO 290
130 O=INT(RND*6)+1
140 S$=S$+"$"+STR$(R)
150 ON O GOTO 160,180,210,240,210,240
160 R=R+N(I):S$=S$+"+"+STR$(N(I))
170 GOTO 270
180 R=R-N(I):IF R<1 THEN 40
190 S$=S$+"-"+STR$(N(I))
200 GOTO 270
210 IF N(I)=1 THEN 40
220 R=R*N(I):S$=S$+"*"+STR$(N(I))
230 GOTO 270
240 IF INT(R/N(I))<>R/N(I) THEN 40
250 IF N(I)=1 THEN 70
260 R=R/N(I):S$=S$+"/"+STR$(N(I))
270 IF R>999 THEN 40
280 S$=S$+"="+STR$(R)
290 NEXT I
300 IF R<100 THEN 40
310 IF NC=7 THEN 500
380 REM ------------------------------
390 REM AUTRES NB
400 FOR I=NC+1 TO 7
410 N(I)=INT(RND*14)+1
420 IF N(I)<11 THEN 440
430 N(I)=(N(I)-10)*25
440 NEXT I
480 REM ------------------------------
490 REM MELANGE
500 FOR I=1 TO 14
510 A=INT(RND*7)+1
520 B=INT(RND*7)+1
530 C=N(A):N(A)=N(B):N(B)=C
540 NEXT I
580 REM ------------------------------
590 REM NB PROPOSES
600 FOR I=1 TO 7
610 PRINT "    ";N(I),
```

```
620 NEXT I
630 PRINT:PRINT
640 PRINT TAB(14);"-->";R:PRINT:PRINT
680 REM --------------------------------
690 REM BOUCLE
700 PRINT CHR$(7);
710 FOR I=1 TO 28500
720 NEXT I
730 PRINT CHR$(7)
780 REM --------------------------------
790 REM SOLUTION
800 PRINT"           MA SOLUTION...":PRINT:PRINT T
AB(11);
810 FOR I=1 TO LEN(S$)
820 IF MID$(S$,I,1)="$" THEN PRINT:PRINT TAB(11);:
GOTO 840
830 PRINT MID$(S$,I,1);
840 NEXT I
850 IF INKEY$<>CHR$(13) THEN 850
860 RUN
```

PROGRAMME

Lignes 30 à 310 : initialisation

La variable NC contient un nombre aléatoire compris entre 4 et 7 donnant le nombre d'opérations à effectuer pour parvenir à la solution. Les chiffres utilisés pour ces opérations sont placés dans le tableau N, l'opération elle-même étant placée dans la chaîne S$ et le résultat dans R.

Dans le cas où la variable R est supérieure à 999 ou inférieure à 100, on retourne au début du programme (ligne 40).

Lignes 400 à 440 : autres chiffres

Si le nombre de chiffres NC est inférieur à 7, on complète le tableau N avec d'autres chiffres aléatoires.

Lignes 500 à 540 : mélange

Pour dérouter le joueur, on mélange le tableau N en utilisant deux nombres aléatoires A et B.

Lignes 600 à 730 : nombres proposés et boucle d'attente

Après l'affichage des 7 chiffres, on effectue une boucle fermée (lignes 710 et 720) d'environ 50 secondes, le temps que le joueur trouve une solution.

Lignes 800 à 860 : solution

La boucle d'attente étant terminée, on affiche à l'écran la solution de l'ordinateur en découpant la chaîne de caractères S$ avant de retourner à une nouvelle proposition dès qu'une touche est enfoncée (ligne 850).

LISTE DES VARIABLES :

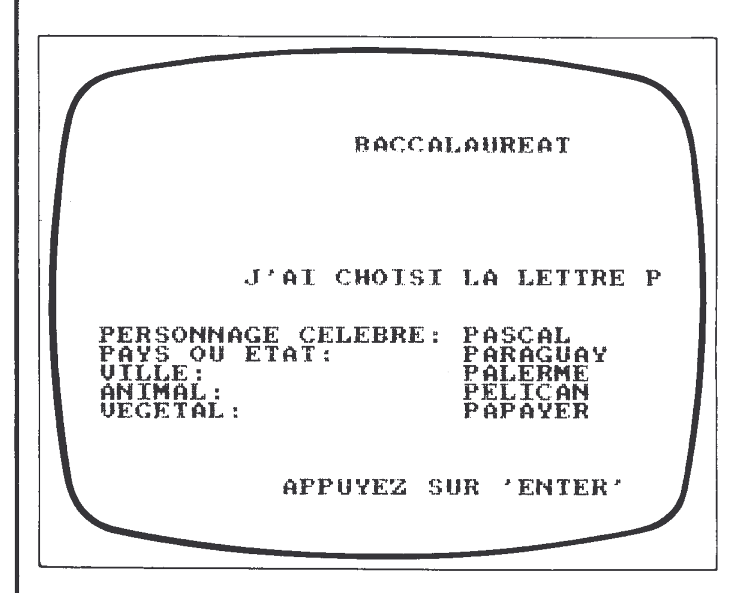
A et **B**	nombre aléatoire pour le mélange
C	utilisée pour les inversions dans le mélange
I	indice de boucle FOR/NEXT
NC	nombre de chiffres utilisés pour le nombre à trouver
N()	tableau des chiffres aléatoires
O	type d'opération à effectuer
R	résultat des opérations
S$	liste des opérations effectuées pour trouver la solution



BACCALAUREAT

Jouez au baccalauréat avec votre Amstrad !

L'ordinateur choisit pour vous une lettre au hasard. Vous devez trouver des noms de personnages célèbres, de pays, de villes, d'animaux et de végétaux commençant par cette lettre. Après le temps qu'il vous conviendra, et dès que vous aurez appuyé sur la touche ENTER, le programme propose ses solutions parmi les 650 mots déjà dans la machine.



```
10 REM BACCALAUREAT
20 REM ------------------------------
30 MODE 1:INK 0,1:INK 1,24:PAPER 0:PEN 1:BORDER 1
40 DEFINT A-Z
50 DIM N$(26,5,5)
60 RANDOMIZE TIME
70 LOCATE 10,10:PRINT"JE M'INSTRUIS ..."
80 LOCATE 2,14:PRINT"LETTRE:";
90 FOR I=1 TO 26
100 PRINT CHR$(I+64);
110 FOR J=1 TO 5
120 FOR K=1 TO 5
130 READ N$(I,J,K)
140 IF ASC(N$(I,J,K))<>I+64 AND N$(I,J,K)<>"*" THE
N LOCATE 0,20:PRINT"ERREUR DANS LES DATAS A LA LET
TRE ";CHR$(I+64):STOP
150 NEXT:NEXT:NEXT
180 REM ------------------------------
190 REM LE JEU
200 L=INT(RND*26)+1
210 CLS:LOCATE 15,2:PRINT"BACCALAUREAT"
220 LOCATE 9,9:PRINT"J'AI CHOISI LA LETTRE ";CHR$(
L+64)
230 PRINT:PRINT:PRINT"PERSONNAGE CELEBRE:":PRINT"P
AYS OU ETAT:"
240 PRINT"VILLE:":PRINT"ANIMAL:":PRINT"VEGETAL:"
250 LOCATE 1,20:PRINT"APPUYEZ SUR 'ENTER' POUR MA
SOLUTION"
260 IF INKEY$<>CHR$(13) THEN 260
270 FOR I=1 TO 5
280 LOCATE 21,I+11
290 N=INT(RND*5)+1
300 IF L=24 AND I=2 THEN PRINT"PAS DE PAYS EN X":G
OTO 330
310 IF N$(L,I,N)="*" THEN 290
320 PRINT N$(L,I,N)
330 NEXT
340 LOCATE 1,20:PRINT CHR$(18);TAB(11);"APPUYEZ SU
R 'ENTER'"
350 IF INKEY$<>CHR$(13) THEN 350
360 GOTO 200
480 REM ------------------------------
490 REM DONNEES
500 DATA ADAM,ADER,ANDERSEN,ANQUETIL,ARAGON
510 DATA AFGHANISTAN,ALGERIE,ALBANIE,ALLEMAGNE,ARG
ENTINE
520 DATA ABIDJAN,ACAPULCO,AIME,AJACCIO,ALGER
530 DATA AIGLE,AIGRETTE,ALLIGATOR,ANTILOPE,ARA
540 DATA ACONIT,AMANITE,ANEMONE,ARACHIDE,ARTICHAUT
550 DATA BACH,BALZAC,BARTOK,BAUDELAIRE,BLERIOT
560 DATA BULGARIE,BELGIQUE,BRESIL,BOLIVIE,BIRMANIE
```

```
570 DATA BADEN-BADEN,BAGDAD,BALE,BAYEUX,BRUXELLES
580 DATA BADOUIN,BALEINE,BAR,BASSET,BELETTE
590 DATA BANANE,BAOBAB,BRUYERE,BRUGNON,BAZZIA
600 DATA CALIGULA,CALVIN,CAMUS,CARNOT,CHURCHILL
610 DATA CAMBODGE,CAMEROUN,CANADA,CHILI,CHINE
620 DATA CAHORS,CALAIS,CASABLANCA,CHAMBERY,CONCARN
EAU
630 DATA CHAT,CAILLE,CHEVAL,CHEVREUIL,CONDOR
640 DATA CACAOYER,CACTUS,CAMELIA,CERISE,CHOUX
650 DATA DAGOBERT,DAGUERRE,DANTON,DENEUVE,DESCARTE
S
660 DATA DANEMARK,DAHOMEY,DOMINICAINE (REP.),DAGHE
STAN,*
670 DATA DAKAR,DELHI,DIEPPE,DJIBOUTI,DOUBAI
680 DATA DAIN,DAUPHIN,DOGUE,DALMATIEN,DINOSAURE
690 DATA DATTE,DAHLIA,DIONEE,DYTIQUE,DAPHNEE
700 DATA EDISON,EIFFEL,EINSTEIN,ELUARD,EURIPIDE
710 DATA ETHIOPIE,ESPAGNE,EQUATEUR,ETATS-UNIS,EGYP
TE
720 DATA EVIAN,EVREUX,EDIMBOURG,ETRETAT,ETAMPE
730 DATA ECREVISSE,EPINOCHE,ESCARGOT,ECUREUIL,ELEP
HANT
740 DATA EPICEA,EPINARD,EGLANTIER,ELLEBORE,ENDIVE
750 DATA FANGIO,FARADAY,FAURE,FELLINI,FERNANDEL
760 DATA FINLANDE,FRANCE,*,*,*
770 DATA FLORENCE,FALAISE,FATIMA,FRANCFORT,FERRARE
780 DATA FAISAN,FAON,FAUCON,FAUVETTE,FURET
790 DATA FUSAIN,FRAISE,FIGUE,FUCHSIA,FOUGERE
800 DATA GABIN,GAGARINE,GALLIENI,GALILEE,GAMA
810 DATA GABON,GHANA,GRECE,GUATEMALA,GUYANE
820 DATA GRENADE,GDANSK,GENES,GENEVE,GLASGOW
830 DATA GAZELLE,GELINOTTE,GERBOISE,GIBBON,GIRAFE
840 DATA GAINIER,GIROFLEE,GLAIEUL,GLYCINE,GRENADIE
R
850 DATA HIPPOCRATE,HINAULT,HITCHCOK,HOMERE,HESSE
860 DATA HONGRIE,HONDURAS,HAWAI,HAITI,HONG-KONG
870 DATA HAMBOURG,HANOI,HELSINKI,HIROSHIMA,HAIPHON
G
880 DATA HAMSTER,HANNETON,HIBOU,HIRONDELLE,HOMARD
890 DATA HARICOT,HETRE,HEVEA,HELIANTHE,HELIOTROPE
900 DATA INGRES,IONESCO,ISABEY,INDY,ICARE
910 DATA IRLANDE,INDE,ISRAEL,ITALIE,ISLANDE
920 DATA ISTANBUL,IRKOUTSK,IBIZA,ISPAHAN,IZMIR
930 DATA IBIS,IDE,IGUANODON,IMPALA,IXODE
940 DATA ICAQUIER,IGNAME,IRIS,IPECA,IPOMEE
950 DATA JACQUARD,JARRY,JAURES,JUNOT,JENSEN
960 DATA JAPON,JORDANIE,JAMAIQUE,JERSEY,JUILAND
970 DATA JAFFA,JERUSALEM,JOHANNESBURG,JEMAPPES,JOH
NSTOWN
980 DATA JAGUAR,JABIRU,JACASSE,JACO,JARS
990 DATA JOUBARDE,JASMIN,JACINTHE,JUTE,JONQUILLE
```

```
1000 DATA KENNEDY,KAFKA,KAPLAN,KEATON,KESSEL
1010 DATA KENYA,KOWEIT,KANSAS,KAZAKHSTAN,KIRGHIZIE
1020 DATA KABOUL,KIEV,KARACHI,KATMANDOU,KIROV
1030 DATA KOALA,KOB,KANGOUROU,KINKAJOU,KERABAU
1040 DATA KAKI,KAPOKIER,KARITE,KAWA,KOLATIER
1050 DATA LAMARTINE,LULLI,LUTHER,LISZT,LAFITTE
1060 DATA LAPONIE,LAOS,LIBAN,LIBERIA,LYBIE
1070 DATA LASSA,LAS VEGAS,LILLE,LISBONNE,LONDRES
1080 DATA LEZARD,LANGOUSTE,LIEVRE,LAMA,LOUTRE
1090 DATA LIN,LYS,LIERRE,LENTILLE,LAITUE
1100 DATA MUSSET,MAHOMET,MOZART,MAURIAC,MARX
1110 DATA MEXIQUE,MAURITANIE,MALDIVES,MALAYSIE,MON
ACO
1120 DATA MADRID,MUNICH,MELBOURNE,MAYENCE,MILAN
1130 DATA MANCHOT,MORUE,MOUCHE,MAINATE,MOULE
1140 DATA MUGUET,MACHE,MAIS,MANDARINE,MOUTARDE
1150 DATA NELSON,NIETZSCHE,NOBEL,NICOT,NEWTON
1160 DATA NIGER,NORVEGE,NICARAGUA,NEPAL,NIGERIA
1170 DATA NAPLES,NOUMEA,NAIROBI,NAGASAKI,NANTES
1180 DATA NAJA,NANDOU,NAUTILE,NEPE,NEREIDE
1190 DATA NAVET,NARCISSE,NEFLE,NENUPHAR,NOIX
1200 DATA OFFENBACH,OHM,OBERKAMPF,OLIVARES,OUDINOT
1210 DATA OUGANDA,OREGON,OHIO,ONTARIO,OKLAHOMA
1220 DATA OTTAWA,OSTENDE,OSLO,OSAKA,ORLEANS
1230 DATA OKAPI,ORANG-OUTANG,ORVET,OTARIE,OURS
1240 DATA OSEILLE,ORTIE,ORCHIDEE,ORANGE,OLIVE
1250 DATA PAGNOL,PAPIN,PARMENTIER,PASCAL,PASTEUR
1260 DATA PAKISTAN,PANAMA,PARAGUAY,PEROU,PHILIPPIN
ES
1270 DATA PALERME,PARIS,PARME,PEKIN,PHILADELPHIE
1280 DATA PANDA,PAON,PANTHERE,PELICAN,PERCHE
1290 DATA PALMIER,PAPAYER,PASTEQUE,PAVOT,PERVENCHE
1300 DATA QUINET,QUILLARD,QUINTANA,QUINTON,QUICHER
AT
1310 DATA QATAR,QUBEC,*,*,*
1320 DATA QUETTA,QUEZON CITY,QUIBERON,QUITO,QUINTI
N
1330 DATA QUELEA,QUISCALE,QUETZAL,*,*
1340 DATA QUASSIA,QUATRE-EPICES,QUEBRACHO,QUETSCHE
,QUINQUINA
1350 DATA RABELAIS,RACINE,REMBRANDT,RICHELIEU,ROSS
INI
1360 DATA RHODESIE,REUNION,ROUMANIE,RUANDA,RUSSIE
1370 DATA RABAT,ROCAMADOUR,ROME,ROUEN,REIMS
1380 DATA RAIE,RAINETTE,RAT,RENNE,ROUSSETTE
1390 DATA RAISIN,RAPHIA,ROSE,RHUBARBE,RICIN
1400 DATA SADE,SARTRE,SCHUBERT,SHAKESPEARE,STALINE
1410 DATA SUEDE,SALVADOR,SAINT-MARIN (REP.),SENEGA
L,SOMALIE
1420 DATA SAIGON,SALZBOURG,SANTIAGO,SAUMUR,STOCKHO
LM
```

```
1430 DATA SAJOU,SALAMANDRE,SANGLIER,SARDINE,SAUTER
ELLE
1440 DATA SALSIFIS,SAUGE,SAULE,SEIGLE,SEQUOIA
1450 DATA TCHAIKOVSKI,TELEMANN,TIBERE,TROTSKI,THEM
ISTOCLE
1460 DATA TANZANIE,TCHAD,TCHECOSLOVAQUIE,THAILANDE
,TURQUIE
1470 DATA TOMBOUCTOU,TOULOUSE,TOURNUS,TRIPOLI,TOUR
S
1480 DATA TORTUE,TAPIR,TARENTULE,TATOU,TAUPE
1490 DATA TOMATE,TREFLE,TROENE,TULIPE,TABAC
1500 DATA ULYSSE,UREY,UTAMARO,URREDA,URRAQUE
1510 DATA URUGUAY,URSS,UKRAINE,USA,UTAH
1520 DATA UTRECHT,USTER,UZEL,URRUGNE,URUGUAIANA
1530 DATA URUBU,UROCYON,URANIE,URANOSCOPE,*
1540 DATA UPAS,URUNDAY,USNEE,*,*
1550 DATA VAILLANT,VALERY,VAUBAN,VAUVENARGUES,VERD
I
1560 DATA VENEZUELA,VIET-NAM,*,*,*
1570 DATA VIENNE,VENISE,VANCOUVER,VERDUN,VINCENNES
1580 DATA VANNEAU,VIPERE,VARAN,VAUROUR,VIGOGNE
1590 DATA VICTORIA-REGIA,VIGNE,VIOLETTE,VOLUBILIS,
VERVEINE
1600 DATA WAGNER,WATT,WATTEAU,WIDAL,WILDE
1610 DATA WISCONSIN,WYOMING,*,*,*
1620 DATA WORCESTER,WORMS,WINCHESTER,WATERLOO,WASC
HINGTON
1630 DATA WAPITI,WORABEE,*,*,*
1640 DATA WALABA,WENGE,WITLOOF,WOLFIA,*
1650 DATA XENOPHON,XENOCRATE,XERXES,XYLANDER,*
1660 DATA *,*,*,*,*
1670 DATA XANTHI,XERTIGNY,XOCHICALCO,XOCHIMILCO,*
1680 DATA XANCUS,XENOPUS,XERUS,XIPHODON,XIPHOPHORE
1690 DATA XANTHORIA,XANTHOSOMA,XANTHO,*,*
1700 DATA YAMAMOTO,YOUNG,YVON,YUKAWA,YSAYE
1710 DATA YEMEN,YOUGOSLAVIE,*,*,*
1720 DATA YAKIMA,YAMAGATA,YAOUNDE,YDES,YOKOHAMA
1730 DATA YACK,YETI,YOKOHAMA,YORKSHIRE-TERRIER,YOR
KSHIRE
1740 DATA YEUSE,YLANG-YLANG,YOHIMBEHE,YUCCA,*
1750 DATA ZAPATA,ZAVATTA,ZEPPELIN,ZEUS,ZOLA
1760 DATA ZAIRE,ZAMBIE,ZELANDE,ZIMBABWE,ZANZIBAR
1770 DATA ZABRZE,ZACATECAS,ZACATECOLUCA,ZURICH,ZAG
REB
1780 DATA ZAMENIS,ZEBRE,ZEBU,ZIBELINE,ZICRONE
1790 DATA ZACINTHE,ZAMIER,ZEEN,ZINNIA,ZOSTERE
```

PROGRAMME

Lignes 30 à 150 : initialisation

Après l'initialisation des nombres aléatoires, le programme place dans le tableau N$ les 650 mots des DATA.

Lignes 200 à 360 : le jeu

Le nombre aléatoire contenu dans la variable L donne la lettre choisie par l'ordinateur. Le programme affiche les cinq types de mots à trouver.

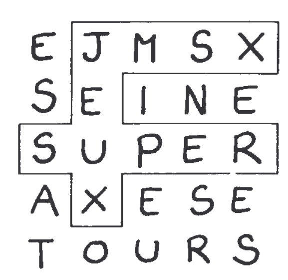
Dès qu'une touche est enfoncée (ligne 260), le programme affiche cinq mots du tableau N$.

Lignes 500 à 1790 : données

Il y a cinq mots par type pour vingt-six lettres. Quand certains mots manquent, ils sont remplacés par des astérisques. Vous pouvez bien sûr ajouter de nouveaux mots.

LISTE DES VARIABLES :

I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
K	indice de boucle FOR/NEXT
L	lettre choisie
N	numéro du mot choisi
N$( )	liste des mots

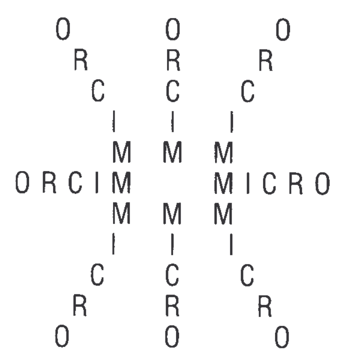


MOTS CACHES

Après avoir donné une liste de mots (de 5 à 15) de 10 lettres maximum, votre ordinateur affiche une grille de 16 lettres sur 16, et à droite de l'écran la liste des mots.

Les mots sélectionnés se trouvent parmi les autres caractères. Vous pouvez les lire à l'endroit (ordinateur) ou à l'envers (ruetanidro) mais non mélangés.

Ils peuvent avoir pris diverses directions, comme le montre le schéma suivant pour le mot MICRO :



Votre ordinateur étant très patient, vous pourrez rester autant de temps que vous voudrez à chercher les mots cachés. Il vous suffira d'appuyer sur ENTER pour obtenir la solution (les lettres qui composent les mots deviennent rouges).

```
10 REM MOTS CACHES
20 REM ------------------------------
30 MODE 1:INK 0,1:INK 1,24:INK 2,6:PAPER 0:PEN 1:B
ORDER 1
40 RANDOMIZE TIME
50 DIM M$(15),G$(20)
60 B$="                    "
70 FOR I=1 TO 20:G$(I)=B$:NEXT I
80 INPUT"NOMBRE DE MOTS";NM
90 IF NM<5 OR NM>15 THEN 80
100 FOR I=1 TO NM
110 PRINT"MOT";I;:INPUT M$(I)
120 IF LEN(M$(I))>12 THEN 110
130 NEXT I
140 FOR I=1 TO NM
150 S=INT(RND+0.5)*-1:IF S=0 THEN S=1
160 D=INT(RND*4)+1
170 X=INT(RND*20)+1
180 Y=INT(RND*20)+1
190 L=LEN(M$(I))
200 ON D GOTO 300,400,500,700
280 REM ------------------------------
290 REM HORIZONTAL
300 P=X+((L-1)*S):IF P<1 OR P>20 THEN 150
310 IF P>X THEN A=X:GOTO 330
320 A=P
330 IF MID$(G$(Y),A,L)<>LEFT$(B$,L) THEN 150
340 C=1:FOR J=X TO P STEP S:G$(Y)=LEFT$(G$(Y),J-1)
+MID$(M$(I),C,1)+RIGHT$(G$(Y),20-J)
350 C=C+1:NEXT J
360 GOTO 710
380 REM ------------------------------
390 REM VERTICAL
400 P=Y+((L-1)*S):IF P<1 OR P>20 THEN 150
410 FOR J=Y TO P STEP S
420 IF MID$(G$(J),X,1)<>" " THEN 150
430 NEXT J
440 C=1:FOR J=Y TO P STEP S:G$(J)=LEFT$(G$(J),X-1)
+MID$(M$(I),C,1)+RIGHT$(G$(J),20-X)
450 C=C+1:NEXT J
460 GOTO 710
480 REM ------------------------------
490 REM HAUT
500 Z=-1
510 P=X+((L-1)*S):IF P<1 OR P>20 THEN 150
520 IF Y+L*Z<1 OR Y+L*Z>20 THEN 150
530 C=Y:A=1:FOR J=X TO P STEP S
540 IF MID$(G$(C),J,1)<>" " THEN 150
550 C=C+Z:A=A+1:NEXT J
560 C=Y:A=1:FOR J=X TO P STEP S
570 G$(C)=LEFT$(G$(C),J-1)+MID$(M$(I),A,1)+RIGHT$(
G$(C),20-J)
580 C=C+Z:A=A+1:NEXT J
```

```
590 GOTO 710
680 REM --------------------------------
690 REM BAS
700 Z=1:GOTO 510
710 NEXT I
780 REM --------------------------------
790 REM EDITION DE LA GRILLE
800 CLS
810 FOR I=1 TO 20
820 FOR J=1 TO 20
830 C$=MID$(G$(I),J,1):IF C$=" " THEN PRINT CHR$(R
ND*25+65);:GOTO 850
840 PRINT C$;
850 NEXT J
860 IF I<=NM THEN PRINT"     ";M$(I);
870 PRINT:NEXT I
880 PRINT:PRINT"APPUYER SUR 'ENTER' POUR LA SOLUTI
ON";
890 IF INKEY$<>CHR$(13) THEN 890
900 PEN 2
910 FOR I=1 TO 20
920 FOR J=1 TO 20
930 IF MID$(G$(I),J,1)<>" " THEN LOCATE J,I:PRINT
MID$(G$(I),J,1)
940 NEXT J,I
950 PEN 1:LOCATE 1,22:PRINT"APPUYER SUR 'ENTER' PO
UR UN AUTRE JEU";
960 IF INKEY$<>CHR$(13) THEN 960
970 RUN
```

LISTE DES VARIABLES :

A	position de début des mots dans la grille G$
A$	dernière touche enfoncée
C	pointeur des caractères dans les mots cachés
C$	caractère à afficher
D	choix du sens d'écriture : horizontal, vertical ou diagonal
G$()	grille contenant les mots cachés
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
L	longueur des mots
M$()	mots choisis
NM	nombre de mots
P	position des mots cachés dans la grille G$()
S	sens d'écriture des mots
X et **Y**	coordonnées d'un caractère dans la grille G$()
Z	Z=1 pour la diagonale vers le bas, Z=−1 pour la diagonale vers le haut

PROGRAMME

Lignes 30 à 200 : initialisation

On saisit le nombre de mots au clavier. Il doit être compris entre 5 et 15 (ligne 90).

Le nombre aléatoire D donne la direction dans laquelle les mots seront écrits dans la grille, la variable S en donnant le sens et X et Y le point de départ.

Lignes 300 à 710 : horizontale, verticale et diagonale (haut et bas)

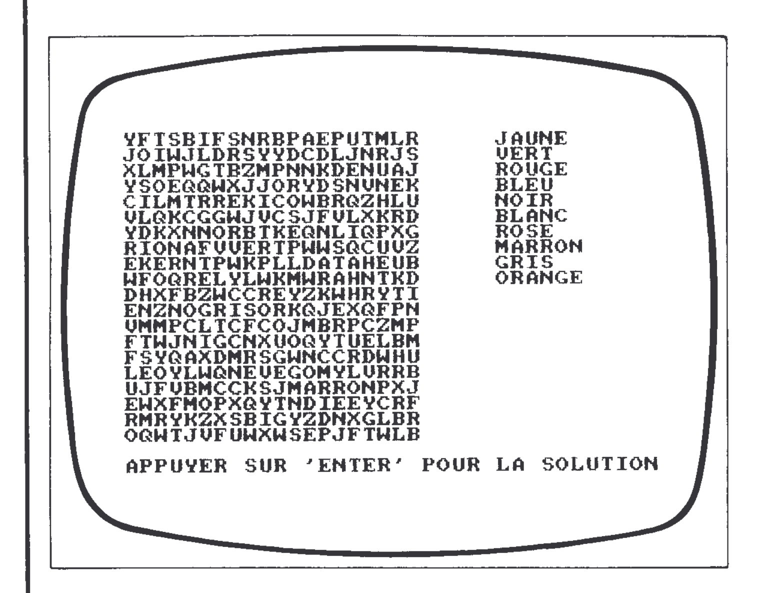
Les nombres aléatoires X et Y désignant l'emplacement du premier caractère à placer dans la grille, on calcule le pointeur P en tenant compte de la longueur et du sens d'écriture (variables L et S).

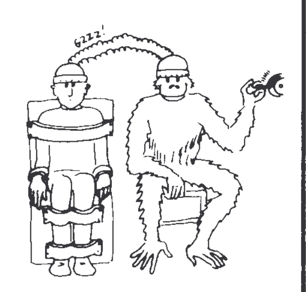
On place chaque caractère du mot M$() dans les cases correspondantes du tableau G$().

Lignes 800 à 970 : résultats

On affiche sur l'écran la grille G$() en remplaçant tous les espaces par des caractères aléatoires compris entre A et Z

Dès que la touche ENTER est appuyée, on passe en revue le tableau G$ pour identifier sur l'écran les mots cachés.





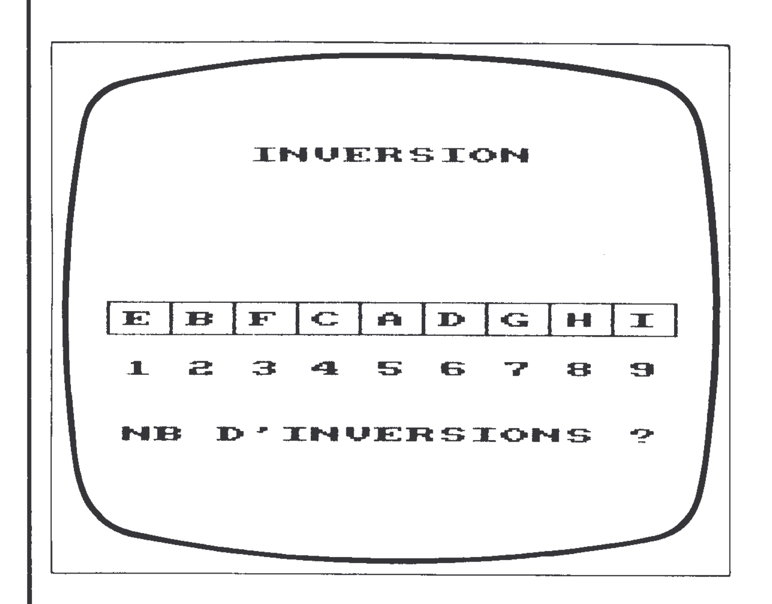
INVERSION

Ce jeu consiste à remettre dans l'ordre les chiffres 1 2 3 4 5 6 7 8 9 ou les lettres A B C D E F G H I que l'ordinateur affiche au départ dans le désordre.

Pour cela il propose d'inverser un nombre de chiffres que vous lui indiquerez à partir de la gauche de la série affichée. Il retourne alors tous les chiffres concernés (ou les lettres) et affiche la nouvelle suite obtenue.

Le renversement se poursuit jusqu'à la remise en ordre des 9 chiffres ou lettres. Le programme affiche alors le nombre d'essais.

A vous de trouver la meilleure tactique !



```
10 REM INVERSION
20 REM ------------------------------
30 MODE 0:RANDOMIZE TIME
40 T=INT(RND*2)+1
50 A$(1)="ABCDEFGHI"
60 A$(2)="123456789"
70 T$=A$(T)
180 REM ------------------------------
190 REM MELANGE
200 FOR I=1 TO 20
210 V=INT(RND*9)+1
220 W=INT(RND*9)+1
230 X$=MID$(T$,V,1)
240 T$=LEFT$(T$,V-1)+MID$(T$,W,1)+RIGHT$(T$,9-V)
250 T$=LEFT$(T$,W-1)+X$+RIGHT$(T$,9-W)
260 NEXT I
280 REM ------------------------------
290 REM DESSIN DU JEU
300 CLS:PRINT TAB(6);"INVERSION"
310 FOR I=80 TO 592 STEP 64
320 PLOT 16,248:DRAW I,248:DRAW I,216:DRAW 16,216:
DRAW 16,248
330 NEXT I
340 FOR I=1 TO 9
350 LOCATE I*2,14:PRINT MID$(STR$(I),2)
360 NEXT I
370 FOR I=1 TO 9
380 LOCATE I*2,11:PRINT MID$(T$,I,1)
390 NEXT I
400 IF T$=A$(T) THEN 700
480 REM ------------------------------
490 REM SAISIE
500 LOCATE 2,18:PRINT"NB D'INVERSIONS ?";
510 N$=INKEY$:IF N$="" THEN 510 ELSE N=VAL(N$)
520 IF N<2 OR N>9 THEN 510
530 FOR I=1 TO N/2
540 X$=MID$(T$,I,1)
550 T$=LEFT$(T$,I-1)+MID$(T$,N-I+1,1)+RIGHT$(T$,9-
I)
560 T$=LEFT$(T$,N-I)+X$+RIGHT$(T$,8-N+I)
570 NEXT I
580 E=E+1
590 GOTO 370
680 REM ------------------------------
690 REM FIN
700 LOCATE 2,18:PRINT"VOUS AVEZ TROUVE ":PRINT TAB
(4);"EN";E;"ESSAIS"
710 IF INKEY$<>CHR$(13) THEN 710
720 RUN
```

PROGRAMME

Lignes 30 à 70 : initialisation

Le tableau A$ donne les suites de lettres ou de chiffres utilisées dans le jeu. La variable T donne au hasard une de ces deux suites.

Lignes 200 à 260 : mélange

Le mélange est effectué en prenant deux des chiffres (ou lettres) au hasard dans T$ et en les inversant. En répétant 20 fois ce processus on obtient une suite tout à fait dans le désordre.

Lignes 300 à 400 : dessin du jeu

Après le dessin du cadre, on utilise une boucle pour afficher les caractères contenus dans la variable T$.

Lignes 500 à 590 : saisie

On saisit par INKEY$ le nombre d'inversions à effectuer. On teste si le nombre donné est supérieur à 1 et inférieur à 10 (ligne 520).

La boucle FOR/NEXT en 530 effectue les permutations dans la variable T$.

La ligne 580 incrémente le compteur d'essais E.

Lignes 700 à 720 : fin de partie

Dès que la variable T$ est égale à la suite initiale (tableau A$), on affiche le nombre d'essais avant de recommencer une nouvelle partie.

LISTE DES VARIABLES :

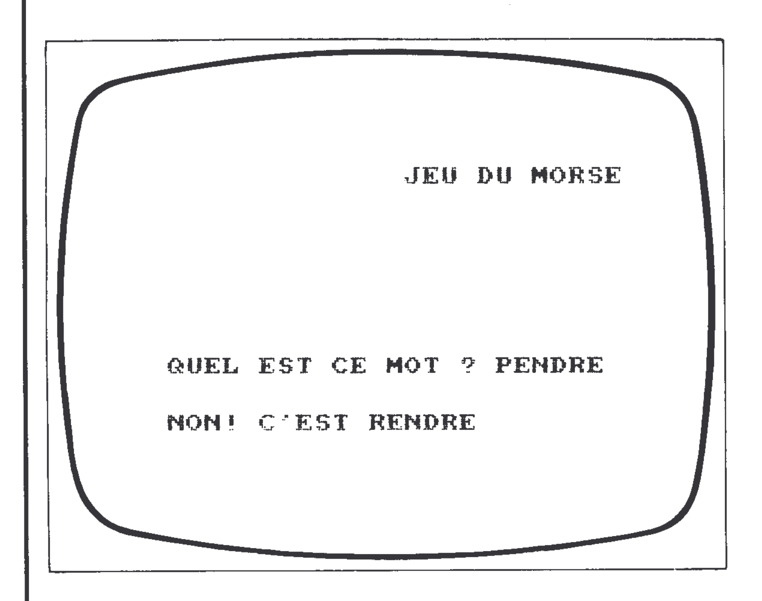
A$	caractère saisi par INKEY$
A$()	liste des caractères à inverser
E	nombre d'inversions
I	indice de boucle FOR/NEXT
N	valeur numérique de N$
N$	nombre d'inversions à effectuer
T	type de liste (chiffres ou lettres)
T$	liste des caractères à inverser
V, W et **X$**	utilisées pour le mélange



JEU DU MORSE

Entraînez-vous au morse tout en vous amusant. L'ordinateur va vous donner en code morse un mot parmi les 300 que contient le programme. A vous de le découvrir en l'écoutant.

Si le mot que vous proposez n'est pas le bon, Amstrad vous donnera la solution. Il suffit d'appuyer sur ENTER pour obtenir une nouvelle proposition.



```
10 REM MORSE
20 REM ----------------------------
30 MODE 1:INK 0,13:INK 1,0:PAPER 0:PEN 1:BORDER 13
40 DEFINT A-Z
50 DIM M$(300)
60 RANDOMIZE TIME
70 FOR I=1 TO 300
80 READ M$(I)
90 NEXT
100 M$=".-  -...-.-.-.. .   ..-.--. ......  .----.
- .-..--  "
110 M$=M$+"-.  --- .--.--.-.-. ... -   ..- ...-.--
 -..--.----.."
180 REM ----------------------------
190 REM LE JEU
200 M=INT(RND*300)+1
210 CLS:LOCATE 14,1:PRINT"JEU DU MORSE"
220 LOCATE 11,11:PRINT"ECOUTEZ CE MOT ..."
230 FOR I=1 TO 6
240 P=1+(ASC(MID$(M$(M),I,1))-65)*4
250 FOR J=P TO P+3
260 IF MID$(M$,J,1)=" " THEN 330
270 IF MID$(M$,J,1)="." THEN 300
280 SOUND 1,63,80,15
290 GOTO 310
300 SOUND 1,63,20,15
310 WHILE SQ(1)>127:WEND
320 FOR K=1 TO 300:NEXT
330 NEXT
340 FOR K=1 TO 1000:NEXT
350 NEXT
360 CLS:LOCATE 14,1:PRINT"JEU DU MORSE"
370 LOCATE 1,11:INPUT"QUEL EST CE MOT ";Z$
380 PRINT:PRINT
390 IF Z$=M$(M) THEN PRINT"OUI! C'EST ";Z$ ELSE PR
INT"NON! C'EST ";M$(M)
400 IF INKEY$<>CHR$(13) THEN 400
410 GOTO 200
480 REM ----------------------------
490 REM LES MOTS
500 DATA ABAQUE,ABBAYE,ABIMER,ABJECT,ABOYER,ABSENT
510 DATA ABSIDE,ABUSER,ACABIT,ACACIA,ACAJOU,ACCENT
520 DATA ACCORD,ACERBE,ACQUIT,ACTEUR,ACTION,ACTUEL
530 DATA ACUITE,ADAGIO,ADORER,ADULER,ADULTE,AERIEN
540 DATA AGACER,AGENCE,AGNEAU,AGREER,AIMANT,ALERTE
550 DATA ALOYAU,AMADOU,AMANDE,ANANAS,ANEMIE,ANGINE
560 DATA ANNEXE,ANODIN,APHONE,APLOMB,APPEAU,AUBURN
570 DATA BADAUD,BALISE,BALZAN,BAROUF,BASSET,BATEAU
580 DATA BAVARD,BECANE,BECTER,BELIER,BERGER,BETAIL
590 DATA BIELLE,BIGLER,BLAGUE,BLEUET,BOHEME,BOUGIE
600 DATA CABRER,CAGEOT,CALICE,CAMPER,CAPTER,CARDAN
```

```
610 DATA CARMIN,CARTON,CASINO,CAVEAU,CENTON,CERCLE
620 DATA CHACUN,CHAPON,CHASTE,CHEQUE,CHOMER,CISEAU
630 DATA CLASSE,CLIENT,CLIQUE,COCHER,COMBLE,CONDOR
640 DATA CONGRE,CONTRE,COPAIN,CORNET,CURARE,CURSIF
650 DATA DANGER,DEBILE,DECHET,DECOTE,DEDALE,DEESSE
660 DATA DELICE,DEMAIN,DEPART,DEPUIS,DESERT,DEVIER
670 DATA DEVOIR,DICTON,DINGUE,DIRECT,DIURNE,DRAGUE
680 DATA ECORCE,ECRIRE,EDITER,ELEVER,EMPIRE,ENCLIN
690 DATA ENIGME,ESCALE,ESCROC,ESPRIT,ETABLE,EXPRES
700 DATA FACHER,FARINE,FAUCON,FELURE,FESTIN,FIGURE
710 DATA FLACON,FLAQUE,FLEUVE,FORMAT,FOUINE,FRAISE
720 DATA FRAUDE,FRELON,FRIAND,FRISER,FUREUR,FURTIF
730 DATA GALANT,GATEAU,GAUFRE,GLAPIR,GOUJON,GOURDE
740 DATA HANCHE,HARGNE,HAUSSE,HARPIE,HUMIDE,HUMOUR
750 DATA IGNARE,ILLICO,IMPACT,INDIGO,INEPTE,INERTE
760 DATA INFAME,INFINI,INTACT,INTIME,IRONIE,IVOIRE
770 DATA JAMAIS,JAMBON,JOCKEY,JONQUE,JOVIAL,JUJUBE
780 DATA KARATE,KERMES,KIMONO,KIRSCH,KLAXON,KOPECK
790 DATA LABEUR,LABOUR,LACUNE,LANGUE,LIASSE,LICITE
800 DATA LIEVRE,LIMITE,LITIGE,LIVIDE,LOUCHE,LUSTRE
810 DATA MACHER,MADERE,MAFFIA,MALICE,MANDAT,MANGUE
820 DATA MANIOC,MARCHE,MARRON,MASQUE,MEDIUM,MEDUSE
830 DATA MEMBRE,MERLIN,MIETTE,MIOCHE,MOTEUR,MOUTON
840 DATA NARINE,NECTAR,NOMBRE,NOTICE,NOUGAT,NUANCE
850 DATA OCCASE,OCTAVE,OFFICE,ORACLE,OTARIE,OUVRIR
860 DATA PAELLA,PAPAYE,PAPIER,PAREIL,PARLER,PATATE
870 DATA PATRIE,PATRON,PAUMER,PAYSAN,PEIGNE,PELOTE
880 DATA PEPITE,PERCHE,PESTER,PIAULE,PILULE,PLANTE
890 DATA POIGNE,POISON,POPOTE,POUPEE,PRETER,PUZZLE
900 DATA QUARTZ,QUATRE,QUICHE,QUILLE,QUINTE,QUITTE
910 DATA RACINE,RADEAU,RAGOUT,RAISIN,RAMURE,RAPHIA
920 DATA RAVIER,REGARD,RELIER,RENDRE,RIFIFI,RUSTRE
930 DATA SAFRAN,SALAMI,SAPEUR,SAVOIR,SCRIPT,SEIGLE
940 DATA SHERIF,SINGER,SIRENE,SOLEIL,SOUPIR,SUIVRE
950 DATA TANDEM,TAPAGE,TAQUIN,TENACE,TESSON,TIMBRE
960 DATA TOPAZE,TOUPIE,TRAHIR,TRICOT,TRIPLE,TULIPE
970 DATA ULCERE,ULTIME,UNIQUE,URBAIN,URGENT,UTOPIE
980 DATA VALIDE,VANTER,VESTON,VIVIER,VOISIN,VOLUME
990 DATA WAPITI,WHISKY,YAOURT,ZENITH,ZIGZAG,ZOUAVE
```

Programme

Lignes 30 à 110 : initialisation

On place dans M$ la liste de codes morse pour tout l'alphabet. Chaque code doit prendre quatre caractères ; des blancs sont ajoutés pour les codes ne comprenant qu'un, deux ou trois points ou tirets. La ligne 60 initialise les nombres aléatoires. La boucle des lignes 70 à 90 place dans le tableau M$ les 300 mots des DATA des lignes 500 à 990.

Lignes 200 à 410 : le jeu

La variable M (nombre aléatoire) pointe le mot que l'on doit trouver.

On édite en code morse par des instructions SOUND les caractères contenus dans M$(M). Chaque caractère est pris séparément (boucle I) et retranscrit en morse (boucle J).

La ligne 370 saisit la réponse du joueur. Si sa solution est incorrecte, la ligne 390 affiche le mot juste.

Lignes 500 à 990 : les mots

Si vous avez déjà tapé le programme « Anagrammes », vous pouvez réutiliser les DATA des lignes 500 à 990 en supprimant les lignes 30 à 340.

Liste des variables :

I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
M	numéro du mot choisi au hasard
M$	liste des codes morse
M$( )	liste des mots
P	pointeur de liste des codes morse
Z$	réponse du joueur



SUPER NOMBRE SECRET

Ce jeu consiste à chercher un nombre secret que l'ordinateur a choisi et qui est compris entre 1 et 99. A chaque solution proposée, le programme répond par un des messages « TROP PETIT » ou « TROP GRAND » par rapport au nombre à trouver.

La première difficulté du jeu est la suivante : votre proposition doit être le résultat d'une opération (exemple : $4 \times 5 = 20$ nombre que vous proposez).

Vous devez débuter avec deux des neuf chiffres de 1 à 9 que vous pouvez multiplier, diviser, additionner ou soustraire. Les opérateurs restent pour tout le jeu les chiffres compris de 1 à 9 ; par contre, vous devez vous servir du résultat de votre première opération pour effectuer la deuxième, puis du résultat de la deuxième pour effectuer la troisième...

Mais la deuxième difficulté du jeu est l'interdiction d'utiliser 2 fois le même chiffre comme opérateur.

Exemple

1er chiffre : 4

1er coup : $4 \times 5 = 20$ « TROP GRAND ». 4&5 ne peuvent être rejoués.

2e coup : $20 \div 2 = 10$ « TROP PETIT» . 4,5&2 ne peuvent être rejoués.

3e coup : $10 + 2 = 12$ « IMPOSSIBLE ». 2 a déjà été joué.

4e coup : $10 + 7 = 17$ « GAGNE EN 3 COUPS ».

Attention, vous n'avez droit qu'à 8 essais pour gagner !

```
10 REM SUPER NB SECRET
20 REM ---------------------------------
30 FOR I=1 TO 9:A(I)=1:NEXT I
40 RANDOMIZE TIME
50 N=INT(RND*99)+1
60 MODE 1:PRINT TAB(11);"SUPER NOMBRE SECRET":PRIN
T:PRINT:PRINT
70 PRINT "IL VOUS RESTE:":PRINT
80 FOR I=1 TO 9
90 IF A(I)=1 THEN PRINT I,
100 NEXT I
110 PRINT:PRINT
180 REM -------------------------------
190 REM LE JEU
200 IF R>0 THEN 240
210 INPUT"PREMIER CHIFFRE";R
220 IF R<1 OR R>9 THEN 210
230 A(R)=0:GOTO 500
240 PRINT"RESULTAT DE L'OPERATION:";R:PRINT
250 INPUT"OPERATION";O$
260 S=VAL(MID$(O$,2))
270 IF S<1 OR S>9 THEN 250
280 IF A(S)=0 THEN 250
290 O$=LEFT$(O$,1)
300 IF O$="*" THEN 370
310 IF O$="+" THEN 390
320 IF O$="-" THEN 410
330 IF O$<>"/" THEN 60
340 IF R/S<1 THEN 60
350 IF INT(R/S)<>R/S THEN 60
360 R=R/S:GOTO 500
370 IF R*S>99 THEN 60
380 R=R*S:GOTO 500
390 IF R+S>99 THEN 60
400 R=R+S:GOTO 500
410 IF R-S<1 THEN 60
420 R=R-S
480 REM -------------------------------
490 REM RESULTAT
500 PRINT:PRINT"RESULTAT:";R:PRINT
510 IF R=N THEN 700
520 IF R<N THEN PRINT"TROP PETIT"
530 IF R>N THEN PRINT"TROP GRAND"
540 IF INKEY$="" THEN 540
550 C=C+1:A(S)=0
560 IF C<9 THEN 60
580 REM -------------------------------
590 REM PERDU
600 PRINT:PRINT"DOMMAGE ! C'ETAIT";N
610 GOTO 710
680 REM -------------------------------
690 REM GAGNE
700 PRINT:PRINT"BRAVO! C'ETAIT";N
710 IF INKEY$<>CHR$(13) THEN 710
720 RUN
```

Programme

Lignes 30 à 110 : initialisation

La ligne 30 remplit le tableau A(1) à A (9) avec le nombre 1 pour connaître les chiffres déjà employés.

La ligne 50 place dans la variable N le nombre secret (nombre aléatoire).

Les lignes 80 à 110 affichent le tableau A(1) à A(9) pour les chiffres restants.

Lignes 200 à 420 : le jeu

Dans le cas où aucune opération n'a été effectuée, on saisit au clavier le premier chiffre.

Sinon on affiche le résultat de la dernière solution proposée, puis on demande l'opération à effectuer et le chiffre à utiliser. La ligne 280 vérifie si vous trichez et renvoie dans ce cas à la saisie de la ligne 250.

Les lignes 290 à 420 recherchent le signe que vous voulez utiliser et effectuent les calculs.

Lignes 500 à 720 : résultats

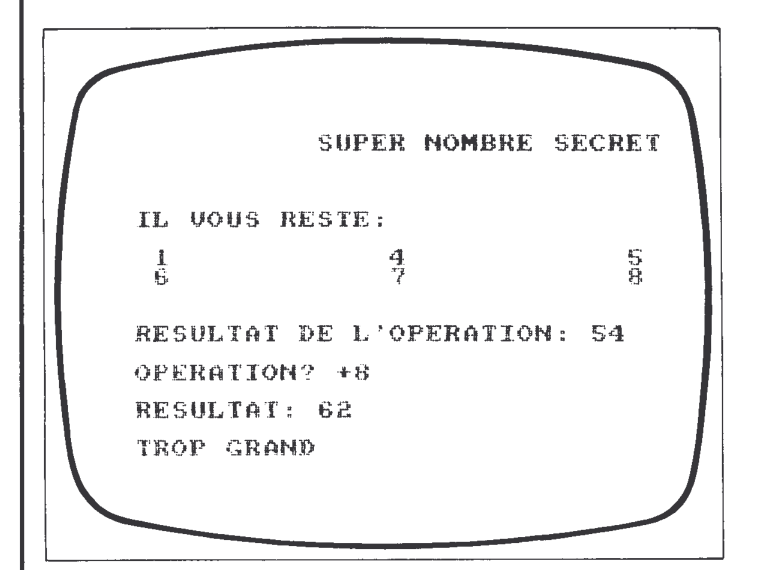
Les lignes 510 à 530 comparent votre solution au nombre secret et renvoient au message correspondant. Si vous avez eu de la chance, la ligne 700 vous félicitera, sinon un message « TROP PETIT » ou « TROP GRAND » apparaîtra.

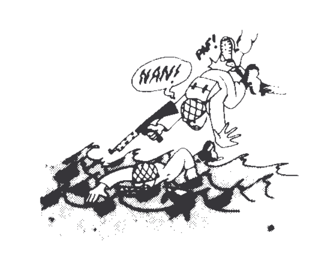
La ligne 550 incrémente le compteur de jeu (variable C) et remet à 0 le chiffre que vous avez utilisé (dans le tableau A()).

Si C=9, on affiche la solution (variable N) puisqu'il ne reste plus de chiffre utilisable.

LISTE DES VARIABLES :

A()	chiffres restants
A$	caractère saisi par INKEY$
C	nombre d'essais
I	indice de boucle FOR/NEXT
N	le nombre secret
O$	opération choisie
R	résultat de la dernière opération
S	chiffre choisi





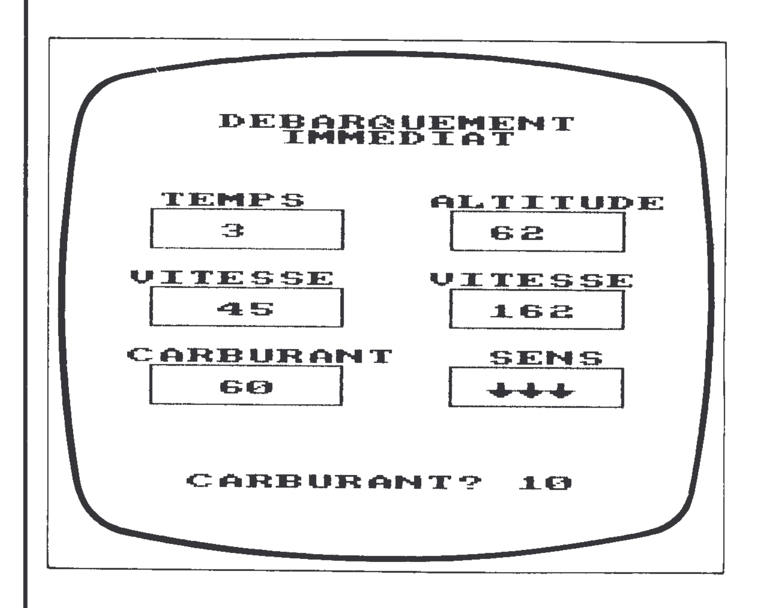
DEBARQUEMENT IMMEDIAT

Poser son vaisseau sur une planète, pour qui est soumis aux dures lois de la pesanteur, n'est pas chose facile.

La vitesse de débarquement peut être maîtrisée en brûlant dans les tuyères du vaisseau une certaine quantité de carburant. Trop de carburant vous fait remonter, pas assez descendre à des vitesses parfois vertigineuses.

Surveillez bien votre altimètre, votre réservoir et bien sûr votre vitesse et c'est un débarquement en douceur.

Mais attention aux arrivées « immédiates » à plus de cinq mètres par seconde !



```
10 REM DEBARQUEMENT IMMEDIAT
20 REM --------------------------------
30 SYMBOL AFTER 128
40 SYMBOL 128,24,24,24,24,255,126,60,24
50 SYMBOL 129,24,60,126,255,24,24,24,24
60 MODE 0:INK 0,1:INK 1,24:INK 2,15:PAPER 0:PEN 1:
BORDER 1
70 A(1)=0:A(2)=200:A(3)=-50:A(4)=180:A(5)=80
80 X(1)=4:Y(1)=8:X(2)=13:Y(2)=8:X(3)=4:Y(3)=13:X(4
)=13:Y(4)=13:X(5)=4:Y(5)=18
90 FOR I=1 TO 5:B(I)=A(I)-1:NEXT I
180 REM ------------------------------
190 REM AFFICHE LE JEU
200 PRINT TAB(5);"DEBARQUEMENT":PRINT TAB(7);"IMME
DIAT"
210 FOR I=8 TO 168 STEP 80
220 PLOT 56,310-I,1:DRAW 260,310-I:DRAW 260,270-I:
DRAW 56,270-I:DRAW 56,310-I:PLOT 376,310-I:DRAW 56
0,310-I:DRAW 560,270-I:DRAW 376,270-I:DRAW 376,310
-I
230 NEXT
240 LOCATE 3,6:PRINT "TEMPS";TAB(12);"ALTITUDE"
250 LOCATE 2,11:PRINT "VITESSE";TAB(12);"VITESSE"
260 LOCATE 2,16:PRINT "CARBURANT";TAB(14);"SENS"
280 REM ------------------------------
290 REM AFFICHE LES DONNEES
300 PEN 2
310 LOCATE 14,18:IF A(3)<0 THEN PRINT CHR$(128)CHR
$(128)CHR$(128) ELSE PRINT CHR$(129)CHR$(129)CHR$(
129)
320 B=0
330 FOR I=1 TO 5
340 IF B(I)=INT(A(I)) OR (I=2 AND B(I)<=0) THEN B=
B+1:GOTO 370
350 IF B(I)<INT(A(I)) THEN B(I)=B(I)+1 ELSE B(I)=B
(I)-1
360 LOCATE X(I),Y(I):PRINT ABS(B(I))
370 NEXT I
380 IF B<5 THEN 320
390 IF B(2)<=0 THEN 700
400 PEN 1
410 IF A(5)=0 THEN R=0:GOTO 560
480 REM ------------------------------
490 REM CALCULS
500 LOCATE 4,24:PRINT CHR$(18);:INPUT"CARBURANT";R
$
510 R=VAL(R$)
520 IF R<0 OR R>A(5) THEN 500
530 IF R>25 THEN R=25
540 A(5)=A(5)-R
```

```
550 IF A(5)=0 THEN LOCATE 3,24:PRINT CHR$(18);"RES
ERVOIR VIDE"
560 A(1)=A(1)+1
570 A(2)=A(2)+A(3)-((5-R)/2)
580 A(3)=A(3)-(5-R)
590 A(4)=ABS(A(3)*3.6)
600 GOTO 300
680 REM -------------------------------
690 REM FIN DE PARTIE
700 IF A(3)<-5 THEN 800
710 LOCATE 1,24:PRINT"DEBARQUEMENT REUSSI"
720 IF INKEY$<>CHR$(13) THEN 720
730 RUN
780 REM ------------------------------
790 REM CRASH...
800 CLS
810 FOR J=1 TO 0 STEP -1
820 FOR I=300 TO 120 STEP -20
830 PLOT I,500-I,J:DRAW 620-I,500-I:DRAW 620-I,I-1
20:DRAW I,I-120:DRAW I,500-I
840 BORDER I MOD 27
850 NEXT I
860 FOR K=1 TO 100:NEXT K
870 NEXT J
880 PEN 1:LOCATE 3,13:PRINT"CRASH A";ABS(A(3));"M/
S"
890 IF INKEY$<>CHR$(13) THEN 890
900 RUN
```

PROGRAMME

Lignes 30 à 260 : initialisation

Après la définition des caractères graphiques, on initialise les variables du tableau de bord (temps, altitude, vitesse et carburant).

Les tableaux X et Y donnent les coordonnées de ces variables sur l'écran.

Les lignes 200 à 260 affichent le dessin du tableau de bord sur l'écran.

Lignes 300 à 410 : affichage des données

En fonction de la vitesse (positive ou négative par rapport au sol), on affiche des flèches vers le bas ou vers le haut comme sens de déplacement (ligne 300).

On affiche à l'aide des tableaux X et Y toutes les coordonnées du tableau de bord en partant du tableau B (anciennes valeurs) pour arriver à celles du tableau A (nouvelles valeurs).

Si l'altitude (B(2)) passe par la valeur 0 pendant la mise à jour, on va directement à la ligne 700 pour la fin de la partie.

Dans le cas où il ne reste plus de carburant (A(5)=0), on saute la ligne 560 pour les calculs sans saisir une quantité de carburant.

Lignes 500 à 600 : calculs

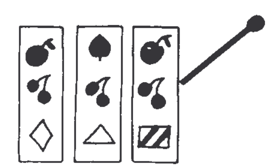
Après la saisie du carburant, on calcule toutes les nouvelles valeurs du tableau de bord.

Si on a utilisé le reste du carburant on affiche en bas de l'écran le message « réservoir vide ».

LISTE DES VARIABLES :

A(1)	temps écoulé
A(2)	altitude
A(3)	vitesse en mètres/seconde
A(4)	vitesse en kilomètres/heure
A(5)	carburant restant
B	nombre de valeurs du tableau B identiques au tableau A
B(1) à **B(5)**	identique au tableau A mais pour les anciennes valeurs
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
K	indice de boucle FOR/NEXT
R$	réponse du joueur
R	valeur numérique de R$
X() et **Y()**	coordonnées sur l'écran des valeurs du tableau de bord

CHAPITRE III : JEUX DE HASARD



JACK-POT

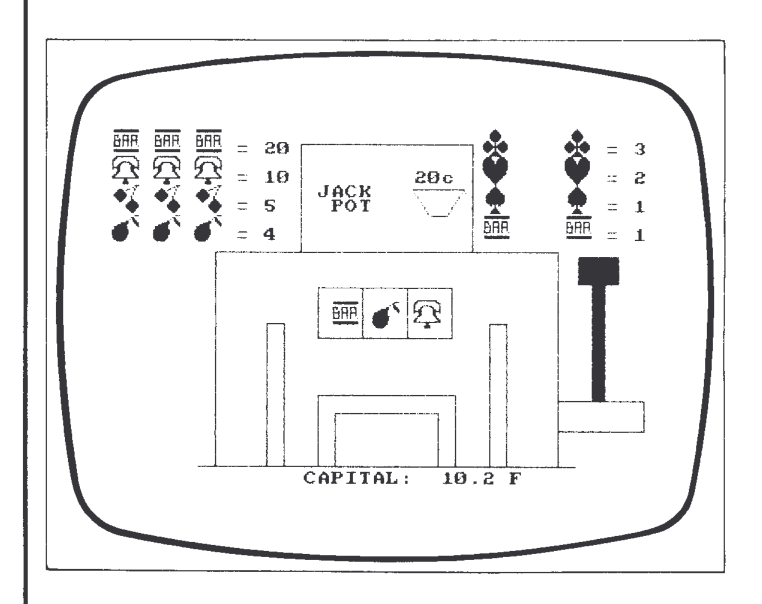
Voilà un jeu qui peut paraître tout d'abord passif puisque le joueur intervient très peu ; mais attention pour le joueur acharné : il risque d'en perdre le sommeil !

Après avoir tapé RUN, le jack-pot se dessine avec les combinaisons gagnantes.

Vous appuyez sur n'importe quelle touche et c'est parti. Quand les symboles s'immobilisent, si vous n'avez rien gagné, votre capital se décrémente de 0,20 franc. Il suffit d'appuyer à nouveau sur n'importe quelle touche pour relancer le jeu.

Quand une combinaison est gagnante, votre gain s'affiche en bas du jack-pot et votre capital augmente d'autant.

Patience, patience, ce jack-pot, digne de ses ancêtres, peut vous prendre beaucoup de temps...



```
10 REM JACK-POT
20 REM ----------------------------
30 SYMBOL AFTER 128:GOSUB 2000
40 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,8:PAPER 0
:PEN 1:GOSUB 3000
50 C=10
60 RANDOMIZE TIME
180 REM ----------------------------
190 REM LE JEU
200 FOR I=1 TO 3
210 D(I)=INT(RND*8)
220 T(I)=INT(RND*15)+5
230 NEXT I
240 GOSUB 1000:GOSUB 1500
250 FOR I=1 TO 3
260 IF T(I)=0 THEN 320
270 T(I)=T(I)-1
280 D(I)=D(I)+1:IF D(I)>7 THEN D(I)=0
290 Z=INT(RND*3)+1
300 PEN Z:LOCATE 14+I*3,13:PRINT CHR$(128+D(I)*4)C
HR$(129+D(I)*4)
310 LOCATE 14+I*3,14:PRINT CHR$(130+D(I)*4)CHR$(13
1+D(I)*4)
320 NEXT I
330 IF T(1) OR T(2) OR T(3) THEN 250
380 REM ----------------------------
390 REM GAINS ?
400 C=INT((C-0.195)*10)/10:CAP=C
410 D=D(1)*100+D(2)*10+D(3)
420 D1=D(1)*100+D(3)
430 IF D=777 THEN C=C+20
440 IF D=666 THEN C=C+10
450 IF D=444 THEN C=C+5
460 IF D=555 THEN C=C+4
470 IF D1=202 THEN C=C+3
480 IF D1=101 THEN C=C+2
490 IF D1=303 THEN C=C+1
500 IF D1=707 THEN C=C+1
510 PEN 2:LOCATE 24,25:PRINT C;"F  ";
520 IF C>CAP THEN 600
530 PRINT CHR$(7)
540 IF INKEY$="" THEN 540
550 IF C=0 THEN 800
560 GOTO 200
580 REM ----------------------------
590 REM GAGNE
600 FOR I=500 TO 50 STEP-5:SOUND 1,I,2:NEXT
610 PEN 2:LOCATE 19,22:PRINT C-CAP
620 IF INKEY$="" THEN 620
630 LOCATE 19,22:PRINT "    "
640 GOTO 200
```

```
780 REM -------------------------------
790 REM PERDU
800 PAPER 0:PEN 1:LOCATE 1,15:PRINT"MERCI":PRINT"P
OUR":PRINT"LES DIX":PRINT"FRANCS"
810 LINE INPUT A$
820 RUN
980 REM -------------------------------
990 REM DESC. POIGNEE
1000 FOR I=10 TO 17
1010 PAPER 0:LOCATE 35,I:PRINT"   "
1020 PAPER 1:LOCATE 35,I+1:PRINT"   "
1030 LOCATE 35,I+2:PRINT"   "
1040 NEXT I
1050 PAPER 0:RETURN
1480 REM ------------------------------
1490 REM MONT. POIGNEE
1500 FOR I=17 TO 10 STEP -1
1510 PAPER 0:LOCATE 35,I+2:PRINT" ";:PAPER 1:PRINT
" ";:PAPER 0:PRINT" "
1520 PAPER 1:LOCATE 35,I+1:PRINT"   "
1530 LOCATE 35,I:PRINT"   "
1540 NEXT I
1550 PAPER 0:RETURN
1980 REM ------------------------------
1990 REM GRAPHIQUES
2000 SYMBOL 128,1,3,7,15,31,63,127,255
2010 SYMBOL 129,128,192,224,240,248,252,254,255
2020 SYMBOL 130,255,127,63,31,15,7,3,1
2030 SYMBOL 131,255,254,252,248,240,224,192,128
2040 SYMBOL 132,56,124,254,254,255,255,255,255
2050 SYMBOL 133,56,124,254,254,254,254,254,254
2060 SYMBOL 134,127,127,63,31,15,7,3,1
2070 SYMBOL 135,252,252,248,240,224,192,128,0
2080 SYMBOL 136,1,3,7,15,15,15,7,51
2090 SYMBOL 137,0,128,192,224,224,224,192,152
2100 SYMBOL 138,121,253,255,253,121,51,7,15
2110 SYMBOL 139,60,126,254,126,60,152,192,224
2120 SYMBOL 140,1,3,7,15,31,63,63,63
2130 SYMBOL 141,0,128,192,224,240,248,248,248
2140 SYMBOL 142,63,63,31,13,1,3,7,15
2150 SYMBOL 143,248,248,240,96,0,128,192,224
2160 SYMBOL 144,0,0,12,30,63,127,127,63
2170 SYMBOL 145,7,26,100,132,8,136,144,16
2180 SYMBOL 146,30,12,0,0,0,0,0,0
2190 SYMBOL 147,24,60,126,255,255,126,60,24
2200 SYMBOL 148,0,0,0,0,0,7,31,127
2210 SYMBOL 149,0,7,24,44,206,199,131,129
2220 SYMBOL 150,127,255,255,255,255,255,127,62
2230 SYMBOL 151,128,192,192,192,128,128,0,0
2240 SYMBOL 152,63,96,192,135,140,152,248,152
2250 SYMBOL 153,252,6,3,225,49,25,31,25
```

```
2260 SYMBOL 154,24,48,96,192,255,1,3,3
2270 SYMBOL 155,24,12,6,3,255,128,192,192
2280 SYMBOL 156,0,127,127,0,115,82,74,123
2290 SYMBOL 157,0,255,255,0,222,82,82,222
2300 SYMBOL 158,74,74,122,0,127,127,0,0
2310 SYMBOL 159,82,82,81,0,255,255,0,0
2320 RETURN
2980 REM ------------------------------
2990 REM DESSIN DE LA MACHINE
3000 FOR I=0 TO 3
3010 READ A,B,C,D:A=A+128:B=B+128
3020 PEN 2:PRINT CHR$(A)CHR$(A+1)" "CHR$(A)CHR$(A+
1)" "CHR$(A)CHR$(A+1);
3030 PEN 3:PRINT TAB(28)CHR$(B)CHR$(B+1)"    "CHR$
(B)CHR$(B+1)
3040 PEN 2:PRINT CHR$(A+2)CHR$(A+3)" "CHR$(A+2)CHR
$(A+3)" "CHR$(A+2)CHR$(A+3)" =";C;
3050 PEN 3:PRINT TAB(28)CHR$(B+2)CHR$(B+3)"    "CH
R$(B+2)CHR$(B+3)" =";D;
3060 NEXT I
3070 PEN 1
3080 PLOT 220,260:DRAW 420,260:DRAW 420,380:DRAW 2
20,380:DRAW 220,260
3090 PLOT 120,20:DRAW 520,20:DRAW 520,260:DRAW 120
,260:DRAW 120,20
3100 PLOT 100,20:DRAW 540,20
3110 PLOT 240,220:DRAW 396,220:DRAW 396,164:DRAW 2
40,164:DRAW 240,220
3120 PLOT 292,220:DRAW 292,164:PLOT 344,220:DRAW 3
44,164
3130 PLOT 350,330:DRAW 410,330:DRAW 390,300:DRAW 3
70,300:DRAW 350,330
3140 PEN 2:LOCATE 23,4:PRINT "20c"
3150 PEN 1:LOCATE 16,5:PRINT "JACK"
3160 PEN 3:LOCATE 17,6:PRINT "POT"
3170 PLOT 520,95:DRAW 620,95:DRAW 620,60:DRAW 520,
60:DRAW 520,95
3180 GOSUB 1500
3190 PAPER 0:PEN 1:LOCATE 15,25:PRINT "CAPITAL:";
3200 PLOT 240,100:DRAW 400,100:DRAW 400,20:DRAW 24
0,20:DRAW 240,100:PLOT 260,80:DRAW 380,80:DRAW 380
,20:DRAW 260,20:DRAW 260,80
3210 PLOT 180,180:DRAW 200,180:DRAW 200,20:DRAW 18
0,20:DRAW 180,180:PLOT 440,180:DRAW 460,180:DRAW 4
60,20:DRAW 440,20:DRAW 440,180
3220 RETURN
3500 DATA 28,8,20,3,24,4,10,2,16,12,5,1,20,28,4,1
```

PROGRAMME

Lignes 30 à 60 : initialisation

L'appel du sous-programme en 2000 détermine les caractères graphiques, et celui en 3000 le dessin complet du jack-pot.

Lignes 200 à 330 : le jeu

Le jeu a besoin des deux tableaux suivants :

■ D() donne le symbole de départ à afficher. Les symboles suivants apparaissent dans l'ordre des caractères graphiques.

■ T() donne le temps, c'est-à-dire le nombre de symboles qui vont défiler sur la machine.

Dès que ces deux tableaux sont définis, on manœuvre la poignée en utilisant les sous-programmes en 1000 et 1500 (respectivement descente et montée), puis on affiche les 3 symboles à partir de D() pour un temps T().

Lignes 400 à 560 : gains ?

On commence par décrémenter le capital C de 20 centimes; puis on place dans CAP le capital restant.

En additionnant D(1) à D(3), on obtient un chiffre qui sert de comparaison avec les solutions gagnantes (variable D). La variable D1 ne prend en considération que les deux extrémités de la combinaison.

On compare D et D1 aux solutions gagnantes et on incrémente le capital (variable C). Quand C est supérieure à CAP on gagne quelque chose. On va alors à la ligne 600 pour afficher le montant des gains ; sinon on retourne à un nouveau jeu.

LISTE DES VARIABLES :

A, B	combinaisons gagnantes
C, D	gains des combinaisons gagnantes
C	capital
CAP	capital avant totalisation des points
D	somme des combinaisons sorties
D1	somme d'une partie des combinaisons
D()	signe à afficher
I	indice de boucle FOR/NEXT
T()	temps de défilement des symboles
Z	nombre aléatoire donnant la couleur des signes



MONTE-CARLO

Voici un jeu de casino qui se joue avec trois dés. Chaque joueur (maximum neuf) doit miser sur une des 29 cases du tapis. Les jetons sont choisis à l'aide des flèches gauche et droite, la touche ENTER validant le choix quand celui-ci est définitif.

C'est le banquier qui lance les dés et paye à chaque joueur un certain nombre de jetons en correspondance avec le tableau suivant :

de I à VI (dés simples) :

on gagne une fois la mise par dé identique. Si l'on joue IV et que deux des trois dés sont des quatre le banquier paye deux fois la mise.

M (manque) :

on gagne une fois la mise si le total des trois dés est inférieur à 11.

de 4 à 17 (somme des dés) :

on gagne si la somme des trois dés est égale au chiffre joué.
4 et 17 : 60 fois la mise
5 et 16 : 30 fois la mise
6 et 15 : 18 fois la mise
7 et 14 : 12 fois la mise
8 et 13 : 8 fois la mise
9 à 12 : 6 fois la mise

P (passe) :

on gagne une fois la mise si le total des trois dés est supérieur à 10.

tous les brelans :

on gagne 30 fois la mise si les trois dés sont identiques.

les brelans :

on gagne 180 fois la mise si les trois dés correspondent à la case jouée.

Comme la partie se joue sur dix tirages, chaque joueur a un capital de 10 jetons dès le début. Le programme affiche les résultats en fin de partie.

```
10 REM MONTE CARLO
20 REM --------------------------------
30 CLS
40 DIM X(29),Y(29),N$(29),G(29),M(29)
50 INPUT"NOMBRE DE JOUEURS ";N
60 IF N<1 OR N>9 THEN PRINT"ENTRE 1 ET 9 !":GOTO 5
0
70 MODE 1:INK 0,1:INK 1 ,24:INK 2,6:INK 3,15:PAPER
 0:PEN 1:BORDER 1
80 TAG:DEG:SYMBOL AFTER 128
90 RANDOMIZE TIME
100 SYMBOL 128,0,0,0,24,24,0,0,0
110 SYMBOL 129,0,24,24,0,0,24,24,0
120 SYMBOL 130,96,96,0,24,24,0,6,6
130 SYMBOL 131,102,102,0,0,0,0,102,102
140 SYMBOL 132,102,102,0,24,24,0,102,102
150 SYMBOL 133,102,102,0,102,102,0,102,102
160 FOR I=1 TO 22:READ X(I),Y(I),N$(I),M(I):NEXT
170 DATA 44,40,I,1,80,40,II,1,116,40,III,1,160,40,
IV,1,204,40,V,1,240,40,VI,1
180 DATA 44,100,M,1,44,170,4,60,44,210,5,30,44,250
,6,18,44,290,7,12
190 DATA 50,330,8,8,76,364,9,6,116,384,10,6,164,38
4,11,6,212,364,12,6,240,330,13,8
200 DATA 244,290,14,12,244,250,15,18,244,210,16,30
,244,170,17,60,244,100,P,1
210 DATA 150,330,30,150,290,180,150,250,180,150,21
0,180,150,170,180,150,130,180,150,90,180
220 FOR I=23 TO 29:READ X(I),Y(I),M(I):NEXT
230 FOR I=1 TO N:P(I)=10:NEXT
480 REM --------------------------------
490 REM DESSIN DU JEU
500 CLG
510 FOR I=0 TO 180 STEP 0.9:MOVE 160,300:PLOTR 120
*COS(I),99*SIN(I),1:NEXT:PLOT 40,300:DRAWR 240,0:D
RAWR 0,-280:DRAWR -240,0:DRAWR 0,280
520 FOR I=0 TO 180:MOVE 160,300:PLOTR 80*COS(I),59
*SIN(I):NEXT:PLOT 80,300:DRAWR 160,0:DRAWR 0,-280:
DRAWR -160,0:DRAWR 0,280
530 PLOT 160,399:DRAW 160,360:PLOT 216,386:DRAW 19
6,352:PLOT 104,386:DRAW 124,352
540 PLOT 258,356:DRAW 228,330:PLOT 62,356:DRAW 92,
330
550 FOR I=120 TO 200 STEP 40
560 PLOT I,60:DRAW I,20
570 NEXT
580 FOR I=260 TO 60 STEP -40
590 IF I<>100 THEN PLOT 40,I:DRAW 280,I ELSE PLOT
80,I:DRAW 240,I
600 NEXT:PLOT 0,400,3
610 FOR I=1 TO 22
```

```
620 FOR J=1 TO LEN(N$(I))
630 MOVE X(I)-8+12*J,Y(I):PRINT MID$(N$(I),J,1);
640 NEXT:NEXT
650 FOR I=1 TO 6:FOR J=1 TO 3
660 MOVE 74+J*40,328-I*40:PRINT CHR$(127+I);
670 PLOT 70+J*40,332-I*40,1:DRAWR 22,0:DRAWR 0,-22
:DRAWR -22,0:DRAWR 0,22:PLOT 120,60:DRAW 120,20:PL
OT 0,400,3
680 NEXT:NEXT
690 MOVE 134,340:PRINT"TOUS";:MOVE 110,320:PRINT"B
RELANS";
700 MOVE 360,380:PRINT"JETONS";
710 FOR I=0 TO 80 STEP 40
720 PLOT 360+I,60:DRAWR 22,0:DRAWR 0,-22:DRAWR -22
,0:DRAWR 0,22
730 NEXT
880 REM ------------------------------
890 REM LE JEU
900 FOR K=1 TO 10
910 ERASE G:DIM G(29)
920 FOR I=0 TO 272 STEP 16:PLOT 520,344-I:PRINT"
    ";:NEXT:PLOT 0,400,2
930 FOR I=1 TO N
940 MOVE 300,370-I*30:PRINT "JOUEUR";RIGHT$(STR$(I
),2);":";P(I);
950 NEXT:PLOT 0,400,1
960 FOR I=1 TO N
970 P=1
980 MOVE 520,370-I*30:PRINT CHR$(243);"    ";:MOVE
 550,370-I*30
990 IF P<23 THEN PRINT N$(P); ELSE IF P>23 THEN PR
INT CHR$(104+P);:MOVE 546,374-I*30:DRAWR 22,0,1:DR
AWR 0,-22:DRAWR -22,0:DRAWR 0,22 ELSE PRINT"TB";
1000 A$=INKEY$:IF A$="" THEN 1000 ELSE IF A$=CHR$(
13) THEN 1050
1010 IF A$=CHR$(242) AND P>1 THEN P=P-1
1020 IF A$=CHR$(243) AND P<29 THEN P=P+1
1030 MOVE 546,374-I*30:DRAWR 22,0,0:DRAWR 0,-22:DR
AWR -22,0:DRAWR 0,22:PLOT 160,300,1
1040 GOTO 980
1050 J(I)=P
1060 NEXT
1070 FOR I=1 TO RND*20+20
1080 FOR J=1 TO 3
1090 D(J)=INT(RND*6)+1
1100 MOVE 324+J*40,56:PRINT CHR$(127+D(J));
1110 NEXT J
1120 NEXT I
1130 D=D(1)+D(2)+D(3)
1140 MOVE 340,20:PRINT"TOTAL:";D;
1150 FOR I=1 TO 6:FOR J=1 TO 3
```

```
1160 IF D(J)=I THEN G(I)=G(I)+1
1170 NEXT:NEXT
1180 IF D<11 THEN G(7)=1
1190 FOR I=8 TO 21
1200 IF D=I-4 THEN G(I)=G(I)+1
1210 NEXT
1220 IF D>10 THEN G(22)=1
1230 IF D(1)<>D(2) OR D(2)<>D(3) THEN 1260
1240 G(23)=1
1250 G(D(1)+23)=1
1260 FOR I=1 TO N
1270 IF G(J(I))<>0 THEN P(I)=P(I)+G(J(I))*M(J(I))
ELSE P(I)=P(I)-1
1280 NEXT
1290 IF INKEY$<>CHR$(13) THEN 1290
1300 NEXT
1480 REM -----------------------------
1490 REM RESULTATS
1500 MODE 1
1510 PRINT TAB(16);"RESULTATS"
1520 FOR I=1 TO N
1530 LOCATE 8,4+I*2:PRINT "JOUEUR";I;": ";USING"##
##";P(I);:PRINT " JETON(S)"
1540 NEXT
1550 IF INKEY$<>CHR$(13) THEN 1550
1560 RUN
```

Programme

Lignes 30 à 230 : initialisation

Après la saisie du nombre de joueurs, le programme initialise les nombres aléatoires (ligne 90).

Les lignes 100 à 150 initialisent les graphiques avec le dessin des six faces d'un dé.

Les lignes 160 à 230 initialisent les tableaux X, Y, N$, M et P (voir liste des variables).

Lignes 500 à 730 : dessin du jeu

On commence par tracer le tapis (lignes 500 à 600), puis on affiche le nom des vingt-deux premières cases (lignes 610 à 640), et pour finir on place les dix-huit dés avec les graphiques correspondants (650 à 680).

Lignes 900 à 1300 : le jeu

La boucle des lignes 930 à 950 affiche le nombre de jetons restant à chaque joueur. Les anciennes valeurs sont effacées par la ligne 920. C'est la boucle des lignes 960 à 1060 qui saisit le choix de chaque joueur (tableau J).

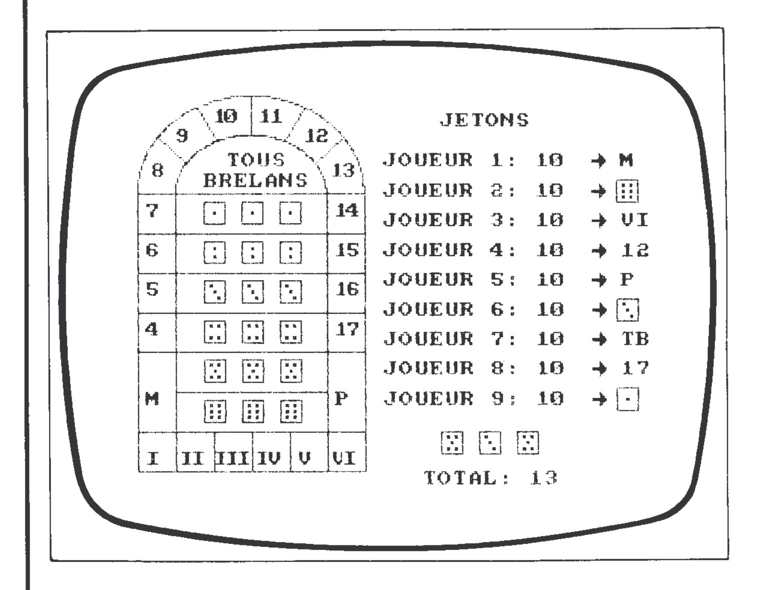
Les dés sont affichés par la boucle des lignes 1070 à 1120 en fonction des nombres aléatoires contenus dans le tableau D.

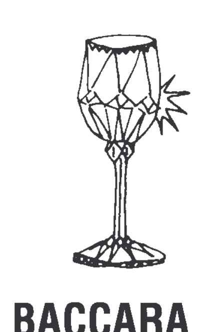
Les lignes 1130 à 1250 recherchent les cases gagnantes. Quand c'est le cas, on incrémente le tableau G.

Il ne reste plus qu'à comparer le tableau G avec celui des joueurs, le tableau J (lignes 1260 à 1280).

LISTE DES VARIABLES :

A$	dernière touche appuyée
D	somme des trois dés
D( )	valeur des trois dés
G( )	cases gagnantes
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
J( )	choix des joueurs
K	indice de boucle FOR/NEXT
M( )	multiplicateur pour les gains
N	nombre de joueurs
N$( )	nom des cases
P	numéro de la case choisie par le joueur
P( )	nombre de jetons restant à chaque joueur
X( ) et **Y( )**	coordonnées des cases sur l'écran





BACCARA

Si vous rêvez de l'ambiance feutrée mais exaltante des casinos, entrez dans le jeu et osez manipuler des sommes fabuleuses.

Un crédit de 10000 francs vous est attribué au début de la partie. La banque – votre adversaire – tire quatre cartes d'un sabot composé de 8 jeux de cinquante-deux cartes. Toutes les cartes conservent leurs valeurs nominales de l'as (1) au neuf ; les figures et le dix ne valent rien. Pour obtenir la valeur totale d'un jeu, on additionne les valeurs de chaque carte en ne retenant que le chiffre des unités.

A chaque donne, l'ordinateur affiche les cartes ; de même, tout au long du jeu, il indique en bas de l'écran les mouvements en cours ainsi que votre mise et vos gains.

La banque donne tout d'abord au ponte deux des quatre cartes et garde les deux autres pour lui.

Vous regardez vos cartes et vous vous trouvez alors devant quatre possibilités pour parler :

■ vous avez 8 ou 9 : c'est-à-dire une main naturelle : c'est au banquier de parler,

■ vous avez 6 ou 7 : vous ne prenez pas d'autre carte,

■ vous avez 5 : vous choisissez ou non de prendre une autre carte,

■ vous avez entre 0 et 4 : vous prenez obligatoirement une autre carte.

Seule la troisième carte que vous aurez pu tirer est visible du banquier.

Puis le banquier parle. Si vous avez 8 ou 9, il faut comparer les jeux. Sinon le banquier a le choix de prendre ou non une carte (choix déterminé par un tableau dans le programme). A ce moment-là, toutes les cartes sont retournées ; le joueur le plus près de 9 gagne la partie.

En tapant sur n'importe quelle touche vous obtenez vos gains et vous pouvez miser pour un autre tour.

Le jeu reprend jusqu'à disparition de tout votre capital...

```
10 REM BACCARA
20 REM ------------------------------
30 SYMBOL AFTER 128:GOSUB 1800
40 MODE 1:INK 0,13:INK 1,0:INK 2,6:INK 3,8:PAPER 0
:PEN 1:BORDER 13
50 DIM J(13,4),S(6,2)
60 J$="123456789XVDR"
70 G=10000
80 RANDOMIZE TIME
90 T$(3)="TTTTTTTTRTT"
100 T$(4)="RRTTTTTTRRT"
110 T$(5)="RRRRTTTTRRT"
120 T$(6)="RRRRRRTTRRR"
130 PEN 1:CLS:LOCATE 10,10:PRINT"JE MELANGE LES CA
RTES"
140 FOR I=1 TO 13
150 FOR K=1 TO 4
160 J(I,K)=0
170 NEXT K,I
180 NC=0
280 REM ------------------------------
290 REM MISE DU PONTE
300 PEN 1:CLS:LOCATE 10,10:PRINT "MISE (MAX";G;"F)
";
310 INPUT M
320 IF M<1 OR M>G THEN 300
380 REM ------------------------------
390 REM TIRAGE DE 6 CARTES
400 FOR I=1 TO 6
410 P(I)=0
420 F=INT(RND*13)+1
430 C=INT(RND*4)+1
440 IF J(F,C)>=8 THEN 420
450 J(F,C)=J(F,C)+1
460 IF F<10 THEN P(I)=F
470 S(I,1)=F:S(I,2)=C
480 NEXT I
490 T=P(1)+P(3)
500 IF T>9 THEN T=T-10
510 TB=P(2)+P(4)
520 IF TB>9 THEN TB=TB-10
580 REM ------------------------------
590 REM AFFICHE LE JEU
600 PEN 3:CLS:PRINT TAB(17);"BACCARA"
610 PEN 2:PRINT:PRINT TAB(18);"PONTE"
620 LOCATE 18,12:PRINT "BANQUE"
630 PEN 3:LOCATE 4,24:PRINT "PONTE:";G;"      MISE
:";M;
680 REM ------------------------------
690 REM CHOIX DU PONTE
700 X=10:Y=4:I=1:GOSUB 1500
```

```
710 X=17:I=3:GOSUB 1500
720 N=0
730 IF T<>9 AND T<>8 THEN 750
740 N=1:GOTO 1020
750 IF T<=5 THEN 800
760 P(5)=10
770 M$="*PONTE NON*"
780 GOSUB 1700
790 GOTO 1000
800 IF T=5 THEN 860
810 M$="*PONTE CARTE*"
820 GOSUB 1700
830 T=T+P(5)
840 IF T>9 THEN T=T-10
850 GOTO 1000
860 LOCATE 13,22:PRINT"CARTE (O/N) ?";
870 A$=INKEY$:IF A$="" THEN 870
880 IF A$="O" THEN 830
890 IF A$<>"N" THEN 870
900 P(5)=10
980 REM ------------------------------
990 REM CHOIX DE LA BANQUE
1000 IF P(5)=10 THEN 1020
1010 X=24:I=5:GOSUB 1500
1020 X=10:Y=13:I=2:GOSUB 1500
1030 X=17:I=4:GOSUB 1500
1040 IF N=1 THEN 1200
1050 IF TB<3 THEN 1110
1060 IF TB>6 THEN 1080
1070 IF MID$(T$(TB),P(5)+1,1)="T" THEN 1110
1080 M$="*BANQUE NON*"
1090 GOSUB 1700
1100 GOTO 1200
1110 M$="*BANQUE CARTE*"
1120 GOSUB 1700
1130 X=24:I=6:GOSUB 1500
1140 TB=TB+P(6)
1150 IF TB>9 THEN TB=TB-10
1180 REM -----------------------------
1190 REM RESULTATS
1200 IF T<>TB THEN 1240
1210 M$="*EXAEQUO*"
1220 GOSUB 1700
1230 GOTO 1320
1240 IF T>TB THEN 1290
1250 M$="*BANQUE GAGNE*"
1260 GOSUB 1700
1270 G=G-M
1280 GOTO 1320
1290 M$="*PONTE GAGNE"+STR$(M)+"F*"
1300 GOSUB 1700
1310 G=G+M
```

```
1320 NC=NC+6
1330 IF INKEY$="" THEN 1330
1340 IF G<1 THEN 1370
1350 IF NC>=204 THEN 130
1360 GOTO 300
1370 PEN 1:CLS
1380 PRINT"MERCI POUR LES DIX":PRINT"MILLE FRANCS.
 LE":PRINT"CASINO ACCEPTE LES":PRINT"CHEQUES!"
1390 END
1480 REM ------------------------------
1490 REM AFFICHE LES CARTES
1500 PEN 1
1510 FOR J=Y+1 TO Y+6:LOCATE X+1,J:PRINT"      ":N
EXT J
1520 PLOT X*16,400-Y*16:DRAW 96+X*16,400-Y*16:DRAW
 96+X*16,304-Y*16:DRAW X*16,304-Y*16:DRAW X*16,400
-Y*16
1530 K=128+(S(I,2)-1)*4
1540 IF S(I,2)>2 THEN Z=1 ELSE Z=2
1550 LOCATE X+2,Y+2:PRINT MID$(J$,S(I,1),1);"   "
1560 PEN Z:LOCATE X+2,Y+3:PRINT " "CHR$(K)CHR$(K+1
)" "
1570 LOCATE X+2,Y+4:PRINT " "CHR$(K+2)CHR$(K+3)" "
1580 PEN 1:LOCATE X+2,Y+5:PRINT "   "MID$(J$,S(I,1
),1)
1590 RETURN
1680 REM ------------------------------
1690 REM AFFICHE UN MESSAGE
1700 PEN 3:LOCATE 20-LEN(M$)/2,22:PRINT M$
1710 FOR I=1 TO 2000:NEXT I
1720 LOCATE 1,22:PRINT CHR$(18)
1730 RETURN
1780 REM ------------------------------
1790 REM GRAPHIQUES
1800 SYMBOL 128,1,3,7,15,31,63,127,255
1810 SYMBOL 129,128,192,224,240,248,252,254,255
1820 SYMBOL 130,255,127,63,31,15,7,3,1
1830 SYMBOL 131,255,254,252,248,240,224,192,128
1840 SYMBOL 132,56,124,254,254,255,255,255,255
1850 SYMBOL 133,56,124,254,254,254,254,254,254
1860 SYMBOL 134,127,127,63,31,15,7,3,1
1870 SYMBOL 135,252,252,248,240,224,192,128,0
1880 SYMBOL 136,1,3,7,15,15,15,7,51
1890 SYMBOL 137,0,128,192,224,224,224,192,152
1900 SYMBOL 138,121,253,255,253,121,51,7,15
1910 SYMBOL 139,60,126,254,126,60,152,192,224
1920 SYMBOL 140,1,3,7,15,31,63,63,63
1930 SYMBOL 141,0,128,192,224,240,248,248,248
1940 SYMBOL 142,63,63,31,13,1,3,7,15
1950 SYMBOL 143,248,248,240,96,0,128,192,224
1960 RETURN
```

PROGRAMME

Lignes 30 à 180 : initialisation

Après la définition des caractères graphiques par le sous-programme en 1800, on place dans le tableau T$ les décisions que prendra le banquier. Chaque liste du tableau contient onze informations. Les dix premières correspondent à la valeur de la carte tirée par le ponte (entre 0 et 9). La onzième précise que le ponte n'a pas pris de carte. La lettre T conseille au banquier de tirer, alors que R l'incite à rester. Chaque ligne correspond au jeu du banquier. Seules sont représentées les solutions pour les valeurs comprises entre 3 et 6 puisque le banquier tire toujours une carte quand il a moins de 3 points et refuse quand il en a plus de 6.

Le tableau J correspond au sabot. Il est remis à zéro quand les cartes sont toutes tirées.

Lignes 300 à 320 : mise du ponte

On saisit la mise du ponte à condition qu'elle ne soit pas supérieure au capital et qu'elle ne soit pas nulle.

Lignes 400 à 520 : tirage des cartes

On tire 6 cartes dans le sabot, F correspondant à la figure de la carte et C à la couleur.On incrémente le tableau J de 1 ; celui-ci ne doit jamais dépasser 8 puisqu'il n'y a que 8 jeux de 52 cartes.

Lignes 700 à 900 : choix du ponte

Après l'affichage des cartes du ponte, celui-ci peut choisir une carte uniquement si le total de ses points est égal à 5.

Dans les autres cas, on suit les règles exposées dans l'introduction de ce chapitre.

Lignes 1000 à 1150 : choix du banquier

Le banquier tire ou non une carte suivant les données du tableau T$ (voir les explications des lignes 30 à 180).

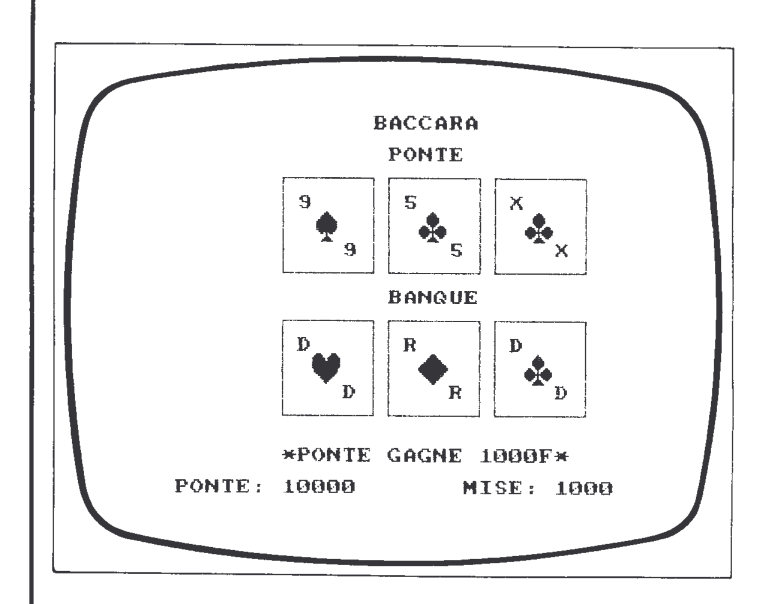
Lignes 1200 à 1390 : résultats et fin

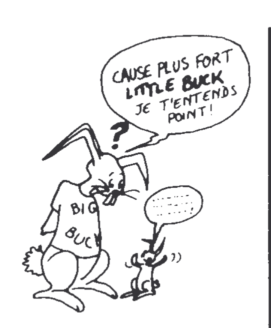
Pour les résultats,il suffit de comparer les jeux du ponte et du banquier (variables T et TB). On les affiche à l'écran ainsi que les gains du ponte. Si ce dernier a toujours de l'argent, on continue la partie.

Dans le cas où le nombre de cartes déjà tirées dépasse 204, on remet le tableau J à zéro avant une nouvelle partie.

LISTE DES VARIABLES :

A$	choix du ponte
C	nombre aléatoire donnant la couleur
F	nombre aléatoire donnant la figure
G	gain du ponte
I	indice de boucle FOR/NEXT
J$	liste des cartes
J(,)	cartes du jeu : 13 cartes et 4 couleurs
K	indice de boucle FOR/NEXT
M	mise du ponte
M$	message à afficher
N	N=1 si main naturelle
NC	nombres de cartes déjà utilisées
P()	valeur des cartes de la partie
S(,)	cartes en jeu
T	total des cartes du ponte
T$()	liste des choix du banquier : – **T** conseille de tirer une carte – **R** incite à rester à 2 cartes
TB	total des cartes du banquier
X et **Y**	coordonnées des cartes sur l'écran





BUCK DICE

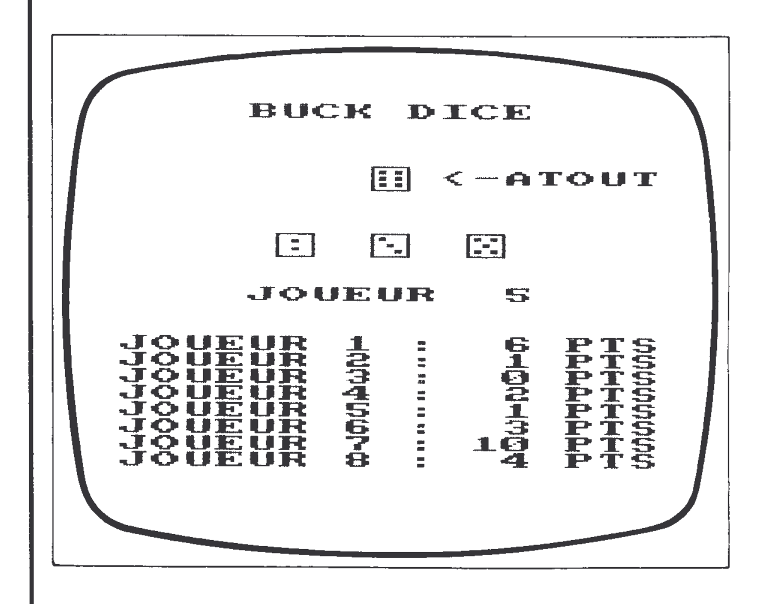
Buck Dice se joue avec quatre dés. Le premier, tiré une seule fois en début de partie, sert d'atout. Chaque joueur lance à tour de rôle les trois autres dés.

Pour gagner une partie, il faut obtenir quinze points. Chaque dé qui correspond à l'atout donne un point. S'il s'agit d'un brelan, appelé Little Buck, on obtient cinq points. Le brelan d'atout, appelé Big Buck, donne automatiquement quinze points quel que soit le total antérieur.

Si un joueur obtient au moins un point il doit rejouer.

A l'exception de Big Buck, il n'est pas possible de dépasser quinze. Le joueur passe donc son tour sans marquer de point dans ce cas.

Pour lancer les dés, chaque joueur doit appuyer sur un des chiffres de un à neuf pour forcer le sort. Après chaque lancer, n'oubliez pas d'appuyer sur ENTER.



```
10 REM BUCK DICE
20 REM ------------------------------------
30 CLS
40 INPUT"NOMBRE DE JOUEURS ";N
50 IF N<1 OR N>8 THEN PRINT"ENTRE 1 ET 8 !":GOTO 4
0
60 MODE 0:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1
70 SYMBOL AFTER 128
80 RANDOMIZE TIME
90 SYMBOL 128,0,0,0,24,24,0,0,0
100 SYMBOL 129,0,24,24,0,0,24,24,0
110 SYMBOL 130,96,96,0,24,24,0,6,6
120 SYMBOL 131,102,102,0,0,0,0,102,102
130 SYMBOL 132,102,102,0,24,24,0,102,102
140 SYMBOL 133,102,102,0,102,102,0,102,102
280 REM ------------------------------
290 REM DESSIN DU JEU
300 PEN 3:LOCATE 5,1:PRINT"BUCK DICE"
310 PEN 2:LOCATE 11,5:PRINT"<-ATOUT"
320 PLOT 254,338:DRAWR 34,0:DRAWR 0,-22:DRAWR -34,
0:DRAWR 0,22
330 FOR I=0 TO 192 STEP 96
340 PLOT 158+I,274:DRAWR 34,0:DRAWR 0,-22:DRAWR -3
4,0:DRAWR 0,22
350 NEXT
360 FOR I=1 TO RND*30+30
370 A=INT(RND*6)+1
380 PEN 3:LOCATE 9,5:PRINT CHR$(127+A)
390 NEXT
480 REM ------------------------------
490 REM LE JEU
500 FOR K=1 TO N
510 AP=P(K)
520 PEN 1
530 LOCATE 5,12:PRINT"JOUEUR ";K
540 Z=VAL(INKEY$):IF Z<1 OR Z>9 THEN 540
550 FOR I=1 TO RND(1)*Z+10
560 PEN 3
570 FOR J=1 TO 3
580 D(J)=INT(RND*6)+1
590 LOCATE 3+J*3,9:PRINT CHR$(127+D(J))
600 NEXT J
610 PEN 1
620 NEXT I
630 IF D(1)<>D(2) OR D(2)<>D(3) THEN 660
640 IF D(1)=A THEN P(K)=15:GOTO 680
650 IF P(K)+5<16 THEN P(K)=P(K)+5
660 P=ABS((D(1)=A)+(D(2)=A)+(D(3)=A))
670 IF P(K)+P<16 THEN P(K)=P(K)+P
680 FOR I=1 TO N
```

```
690 LOCATE 1,14+I:PRINT "JOUEUR";I;":";USING" ## "
;P(I);:PRINT "PTS"
700 NEXT
710 IF INKEY$<>CHR$(13) THEN 710
720 IF P(K)=15 THEN 900
730 IF AP<P(K) THEN 510
740 NEXT:GOTO 500
880 REM -------------------------------
890 REM RESULTATS
900 MODE 1
910 PRINT TAB(15);"RESULTATS"
920 FOR I=1 TO N
930 LOCATE 8,6+I*2:PRINT"JOUEUR";I;":";USING" ## "
;P(I);:PRINT"POINT(S)"
940 NEXT
950 IF INKEY$<>CHR$(13) THEN 950
960 RUN
```

PROGRAMME

Lignes 30 à 140 : initialisation

Après la saisie du nombre de joueurs, le programme initialise les nombres aléatoires (ligne 80).

Les lignes 100 à 140 initialisent les graphiques avec le dessin des six faces d'un dé.

Lignes 300 à 390 : dessin du jeu

Les lignes 320 à 350 dessinent sur l'écran les cadres des quatre dés. C'est la boucle de 360 à 390 qui donne l'atout.

Lignes 500 à 740 : le jeu

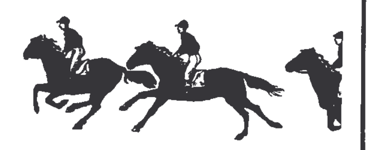
Les dés sont affichés par la boucle des lignes 550 à 620 en fonction des nombres aléatoires contenus dans le tableau D.

Les lignes 630 à 670 calculent les points obtenus par le joueur.

La boucle des lignes 680 à 700 affiche le nombre de jetons restant à chaque joueur.

LISTE DES VARAIBLES :

A	valeur de l'atout
AP	nombre de points d'un joueur avec un coup
D( )	valeur des trois dés
I	indice de boucle FOR/NEXT
K	indice de boucle FOR/NEXT
N	nombre de joueurs
P	nombre de dés correspondant à l'atout
P( )	nombre de jetons restant à chaque joueur
Z	chiffre tapé au clavier par le joueur



TIERCE

Pourquoi vous priver d'assister en direct au tiercé quand il pleut ou qu'il fait froid ?

Branchez vite votre Amstrad, réunissez votre famille, vos amis, et les paris sont ouverts.

Huit joueurs au maximum peuvent inscrire leurs combinaisons de trois chevaux de 1 à 9.

Puis vos favoris n'attendent que l'enfoncement d'une touche pour s'élancer sur l'hippodrome.

Le 3, le 5... non c'est le 7 qui l'emporte d'une longueur ! De toute façon l'ordre d'arrivée s'inscrit au fur et à mesure au bas de l'écran.

Quand la course est finie, vous pouvez taper sur n'importe quelle touche pour avoir les résultats des joueurs.

Pour le tiercé dans l'ordre, vous « touchez » dix points ; dans le désordre 5 points. Pour deux chevaux gagnants vous touchez 2 points et pour un seul cheval rien du tout.

Mais vous pouvez rejouer très vite en tapant sur une touche et aux prochains résultats vous obtiendrez le cumul de vos points dans la colonne total.

Alors, prêts pour une deuxième course ?

```
10 REM TIERCE
20 REM ---------------------------------
30 SYMBOL AFTER 128:GOSUB 2000
40 MODE 1:INK 0,9:INK 1,0:INK 2,6:INK 3,15:PAPER 0
:PEN 1:BORDER 9
50 INPUT"NB DE PARIEURS";PA
60 IF PA>8 THEN 40
70 PRINT:PRINT
80 FOR I=1 TO PA
90 PRINT"NOM DU PARIEUR";I;
100 INPUT N$(I)
110 NEXT I
120 RANDOMIZE TIME
180 REM -----------------------------
190 REM ENTREE DES PARIS
200 C=C+1
210 CLS:PRINT:PRINT TAB(14);"COURSE";C
220 FOR I=1 TO PA
230 PRINT:PRINT
240 FOR J=1 TO 3
250 PRINT"CHEVAL";J;"DE ";N$(I)
260 INPUT C(I,J)
270 IF C(I,J)>9 THEN 250
280 NEXT J,I
380 REM -----------------------------
390 REM LA COURSE
400 CLS:PEN 2:PRINT TAB(14);"COURSE";C:PRINT:PRINT
:PRINT:PEN 1
410 PRINT STRING$(36,"-")
420 FOR I=1 TO 9
430 PEN 3:PRINT I;:PEN 1:PRINT A$(1);CHR$(134);:PE
N 3:PRINT TAB(36);CHR$(133)
440 PRINT "       ";CHR$(134);TAB(36);CHR$(133)
450 NEXT I:PEN 1
460 PRINT STRING$(36,"-")
470 FOR I=1 TO 9:R(I)=0:X(I)=4:NEXT
480 PEN 1
500 IF INKEY$="" THEN 500
510 FOR I=6 TO 23:LOCATE 7,I:PRINT " ":NEXT
600 I=INT(RND*9)+1
610 IF X(I)>36 THEN 600
620 LOCATE X(I),4+I*2:PRINT A$(1):FOR J=1 TO 50:NE
XT J
630 LOCATE X(I),4+I*2:PRINT A$(2)
640 X(I)=X(I)+1
650 FOR J=1 TO 40:NEXT J
660 IF X(I)>=36 THEN 700
670 GOTO 600
680 REM -----------------------------
690 REM VERIF. ARRIVEE
700 X(I)=X(I)+1
```

```
710 NA=NA+1:R(NA)=I
720 PEN 2:LOCATE NA*2,25:PRINT I;:PEN 1
730 IF NA<9 THEN 600
780 REM ------------------------------
790 REM RESULTATS
800 IF INKEY$="" THEN 800
810 CLS:PEN 2:PRINT TAB(12);"RESULTATS":PEN 1
820 PRINT:PRINT"PLACE          NO"
830 FOR I=1 TO 9:PRINT I,R(I):NEXT I
840 PRINT:PRINT"NOM    C1 C2 C3 PTS TOT."
850 FOR I=1 TO PA
860 NO=0:T=0:TP=0
870 FOR J=1 TO 3
880 FOR K=1 TO 3
890 IF R(J)<>C(I,K) THEN 910
900 NO=NO+1
910 IF J<>K OR R(J)<>C(I,K) THEN 930
920 T=T+1
930 NEXT K,J
940 IF T=3 THEN TP=TP+5
950 IF NO=3 THEN TP=TP+5
960 IF NO=2 THEN TP=TP+2
970 P(I)=P(I)+TP
980 PRINT LEFT$(N$(I)+"      ",6);" ";C(I,1);C(I,2)
;C(I,3);TP;" ";P(I)
990 NEXT I
1000 IF INKEY$="" THEN 1000
1010 NA=0
1020 GOTO 200
1980 REM ----------------------------
1990 REM GRAPHIQUES
2000 SYMBOL 128,0,0,0,12,3,3,5,10
2010 SYMBOL 129,8,12,26,113,255,254,1,1
2020 SYMBOL 130,0,128,224,144,192,128,64,32
2030 SYMBOL 131,0,0,1,195,63,63,80,80
2040 SYMBOL 132,128,196,174,153,252,224,20,34
2050 SYMBOL 133,168,80,168,80,168,80,168,80
2060 SYMBOL 134,128,128,128,128,128,128,128,128
2070 A$(1)=CHR$(128)+CHR$(129)+CHR$(130)
2080 A$(2)=" "+CHR$(131)+CHR$(132)
2090 RETURN
```

Programme

Lignes 30 à 120 : initialisation

On saisit au clavier le nombre et le nom des parieurs.

Lignes 200 à 280 : entrée des paris

Après l'incrément du numéro de la course, chaque parieur donne à tour de rôle sa combinaison de trois chevaux. Celle-ci est placée dans le tableau à deux dimensions C(,).

Lignes 400 à 730 : dessin, course et vérification de l'arrivée

Après le dessin de l'hippodrome, le jeu commence en déplaçant aléatoirement les chevaux (variable I).

La ligne 730 teste si tous les chevaux sont sur la ligne d'arrivée.

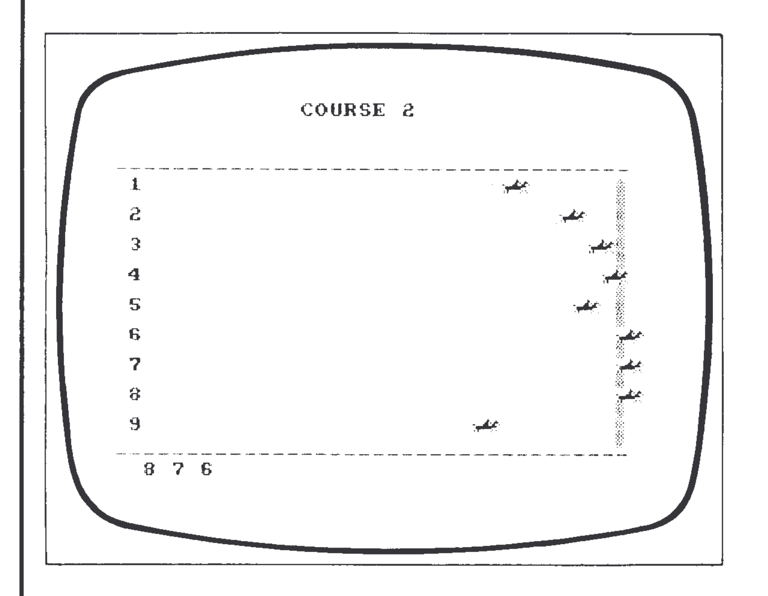
Lignes 800 à 1020 : résultats

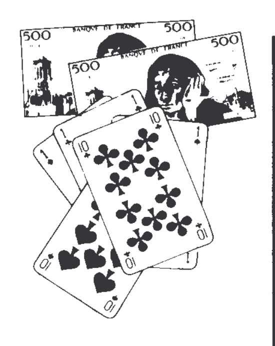
Cette partie du programme donne les résultats à l'arrivée et compte le nombre de points acquis par chaque parieur.

Dès qu'une touche d'Amstrad est appuyée, le programme retourne à la ligne 200 pour une nouvelle course.

LISTE DES VARIABLES :

A$( )	dessin des chevaux
C	numéro de la course
C(,)	chevaux joués par chaque parieur
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
K	indice de boucle FOR/NEXT
N$()	nom des parieurs
NA	numéro d'arrivée
NO	nombre de chevaux dans l'ordre
P()	total des points (cumul de toutes les courses)
PA	nombre de parieurs
R()	tableau d'arrivée
T	nombre de chevaux dans le désordre
TP	total des points de la dernière course
X()	position des chevaux pendant la course





POKER

Après avoir rentré votre mise (au maximum le montant du capital affiché) l'ordinateur vous propose cinq cartes.

A vous de préparer une combinaison en gardant (taper O pour oui) ou en changeant (taper N pour non) les cartes désignées par la flèche.

Les résultats seront les suivants :

Même pas une paire	aucune des combinaisons qui suivent.	mise perdue
1 paire	2 cartes de même valeur	mise perdue
2 paires	2 × 2 cartes de même valeur	rien
Brelan	3 cartes de même valeur	mise gagnée
Séquence	5 cartes qui se suivent dans n'importe quelle couleur	2 fois la mise
Full	Brelan + paire	3 fois la mise
Flush	5 cartes de la même couleur	4 fois la mise
Carré	4 cartes identiques	5 fois la mise
Flush royal	5 cartes se suivant dans la même couleur	6 fois la mise

Pour les cinq dernières combinaisons, l'ordinateur propose de doubler votre mise (répondre O ou N). Si vous acceptez, vous devez lui dire si la prochaine carte affichée est rouge ou noire (répondre R ou N). Bien sûr vous perdez tout si vous vous êtes trompé !

Le capital est alors mis à jour et vous pouvez continuer la partie jusqu'à épuisement de vos fonds.

```
10 REM POKER
20 REM ------------------------------
30 SYMBOL AFTER 128:GOSUB 2000
40 MODE 1:INK 0,13:INK 1,0:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 13
50 DIM J(8,4)
60 J$="789XVDR1"
70 G=100
80 RANDOMIZE TIME
90 FOR I=1 TO 8
100 C(I)=0
110 FOR K=1 TO 4
120 J(I,K)=0
130 NEXT K,I
180 REM ------------------------------
190 REM MISE DU JOUEUR
200 PEN 1:CLS:LOCATE 10,10:PRINT "MISE (MAX";G;"F)
";
210 INPUT M
220 IF M<1 OR M>G THEN 200
280 REM ------------------------------
290 REM TIRAGE DE 10 CARTES
300 FOR I=1 TO 10
310 P(I)=0
320 F=INT(RND*8)+1
330 C=INT(RND*4)+1
340 IF J(F,C)=1 THEN 320
350 J(F,C)=1
360 S(I,1)=F:S(I,2)=C
370 NEXT I
480 REM ------------------------------
490 REM LE JEU
500 PEN 2:CLS:PRINT TAB(19);"POKER"
510 PEN 3
520 LOCATE 4,24:PRINT "CAPITAL:";G;"          MIS
E:";M;
530 C=0
540 FOR I=1 TO 5
550 X=I*7-3:Y=3:GOSUB 1300
560 NEXT I
570 FOR I=1 TO 5
580 LOCATE 1+I*7,12:PRINT "^"
590 A$=INKEY$:IF A$="" THEN 590
600 IF A$="N" THEN 640
610 IF A$<>"O" THEN 590
620 LOCATE 1+I*7,11:PRINT "O"
630 GOTO 660
640 S(I,1)=S(I+5,1):S(I,2)=S(I+5,2)
650 LOCATE 1+I*7,11:PRINT "N"
660 LOCATE 1+I*7,12:PRINT " "
670 NEXT I
```

```
680 FOR I=1 TO 5
690 X=I*7-3:GOSUB 1300
700 NEXT I
710 FOR I=1 TO 8
720 P(I)=0
730 FOR J=1 TO 5
740 IF S(J,1)=I THEN P(I)=P(I)+1
750 NEXT J
760 FOR J=2 TO 4
770 IF P(I)=J THEN C(J)=C(J)+1
780 NEXT J
790 NEXT I
800 IF C(4)=1 THEN C=7:GOTO 940
810 IF C(3)=1 AND C(2)=1 THEN C=5:GOTO 900
820 IF C(3)=1 THEN C=3:GOTO 900
830 IF C(2)=2 THEN C=2:GOTO 900
840 IF C(2)=1 THEN C=1:GOTO 900
850 FOR I=5 TO 8
860 FOR J=1 TO 5
870 IF S(J,1)>I OR S(J,1)<I-4 THEN 890
880 NEXT J:C=4
890 NEXT I
900 FOR I=2 TO 5
910 IF S(I,2)<>S(I-1,2) THEN 940
920 NEXT I
930 IF C=4 THEN C=8 ELSE C=6
940 LOCATE 22-LEN(C$(C))/2,20:PRINT C$(C)
950 IF C<2 THEN 1100
960 IF C<3 THEN LOCATE 8,21:PRINT"DEUX PAIRES C'ES
T PAS ASSEZ":GOTO 1110
970 LOCATE 16,21:PRINT"MISE * PAR";C-2
980 IF C=3 THEN 1090
990 LOCATE 13,22:PRINT "VOUS VOULEZ * PAR";(C-2)*2
1000 A$=INKEY$:IF A$="" THEN 1000 ELSE IF A$<>"O"
THEN 1090 ELSE LOCATE 16,21:PRINT"MISE * PAR";(C-2
)*2
1010 LOCATE 1,22:PRINT"LA PROCHAINE CARTE EST ROUG
E OU NOIRE ?"
1020 A$=INKEY$:IF A$="" THEN 1020 ELSE IF A$<>"N"
AND A$<>"R" THEN 1020
1030 X=18:Y=12:I=1
1040 F=INT(RND*8)+1:Z=INT(RND*4)+1:IF J(F,Z)=1 THE
N 1040 ELSE S(1,1)=F:S(1,2)=Z
1050 GOSUB 1300
1060 IF (S(1,2)<2 AND A$="N") OR (S(1,2)>2 AND A$=
"R") THEN 1080
1070 G=G+M*(C-2)*2:GOTO 1110
1080 G=G-M*(C-2)*2:GOTO 1110
1090 G=G+M*(C-2):GOTO 1110
1100 G=G-M
1110 IF INKEY$="" THEN 1110
```

```
1120 IF G>0 THEN 80
1130 CLS
1140 PRINT"AVEC L'ARGENT QUE VOUS AVEZ"
1150 PRINT"PERDU, JE VAIS M'ACHETER"
1160 PRINT"UNE EXTENSION ...!"
1170 END
1280 REM ------------------------------
1290 REM AFFICHE LES CARTES
1300 PEN 1
1310 FOR J=Y+1 TO Y+6:LOCATE X+1,J:PRINT"      ":N
EXT J
1320 PLOT X*16,400-Y*16:DRAW 96+X*16,400-Y*16:DRAW
 96+X*16,304-Y*16:DRAW X*16,304-Y*16:DRAW X*16,400
-Y*16
1330 K=128+(S(I,2)-1)*4
1340 IF S(I,2)>2 THEN Z=1 ELSE Z=2
1350 LOCATE X+2,Y+2:PRINT MID$(J$,S(I,1),1);"   "
1360 PEN Z:LOCATE X+2,Y+3:PRINT " "CHR$(K)CHR$(K+1
)" "
1370 LOCATE X+2,Y+4:PRINT " "CHR$(K+2)CHR$(K+3)" "
1380 PEN 1:LOCATE X+2,Y+5:PRINT "   "MID$(J$,S(I,1
),1)
1390 RETURN
1980 REM ------------------------------
1990 REM GRAPHIQUES
2000 SYMBOL 128,1,3,7,15,31,63,127,255
2010 SYMBOL 129,128,192,224,240,248,252,254,255
2020 SYMBOL 130,255,127,63,31,15,7,3,1
2030 SYMBOL 131,255,254,252,248,240,224,192,128
2040 SYMBOL 132,56,124,254,254,255,255,255,255
2050 SYMBOL 133,56,124,254,254,254,254,254,254
2060 SYMBOL 134,127,127,63,31,15,7,3,1
2070 SYMBOL 135,252,252,248,240,224,192,128,0
2080 SYMBOL 136,1,3,7,15,15,15,7,51
2090 SYMBOL 137,0,128,192,224,224,224,192,152
2100 SYMBOL 138,121,253,255,253,121,51,7,15
2110 SYMBOL 139,60,126,254,126,60,152,192,224
2120 SYMBOL 140,1,3,7,15,31,63,63,63
2130 SYMBOL 141,0,128,192,224,240,248,248,248
2140 SYMBOL 142,63,63,31,13,1,3,7,15
2150 SYMBOL 143,248,248,240,96,0,128,192,224
2160 C$(1)="UNE PAIRE":C$(2)="DEUX PAIRES":C$(3)="
BRELAN"
2170 C$(4)="SEQUENCE":C$(5)="FULL":C$(6)="FLUSH"
2180 C$(7)="CARRE":C$(8)="FLUSH ROYAL":C$(0)="MEME
 PAS UNE PAIRE"
2190 RETURN
```

Programme

Lignes 30 à 130 : initialisation

Après la définition des caractères graphiques et des combinaisons gagnantes dans le tableau C$ (sous-programme en 2000), on initialise à 0 le tableau J correspondant au jeu de cartes. Il est remis à zéro à chaque partie.

Lignes 200 à 220 : mise du joueur

On saisit la mise du joueur à condition qu'elle ne soit pas supérieure au capital et qu'elle ne soit pas nulle.

Lignes 300 à 370 : tirage de 10 cartes

On tire 10 cartes dans le jeu, F correspondant à la figure de la carte et C à la couleur. On incrémente le tableau J de 1 ; on teste celui-ci à chaque tirage pour ne pas utiliser deux fois la même carte.

Lignes 500 à 1170 : le jeu

Après l'affichage des cartes avec le sous-programme en 1300, on saisit le choix du joueur.

Puis on affiche les nouvelles cartes et l'on suit les règles exposées dans l'introduction de ce chapitre.

Les boucles des lignes 710 à 790 placent dans le tableau C le nombre de cartes identiques (exemple : C(4)=1 donne un carré, C(2)=2 donne deux paires, etc...)

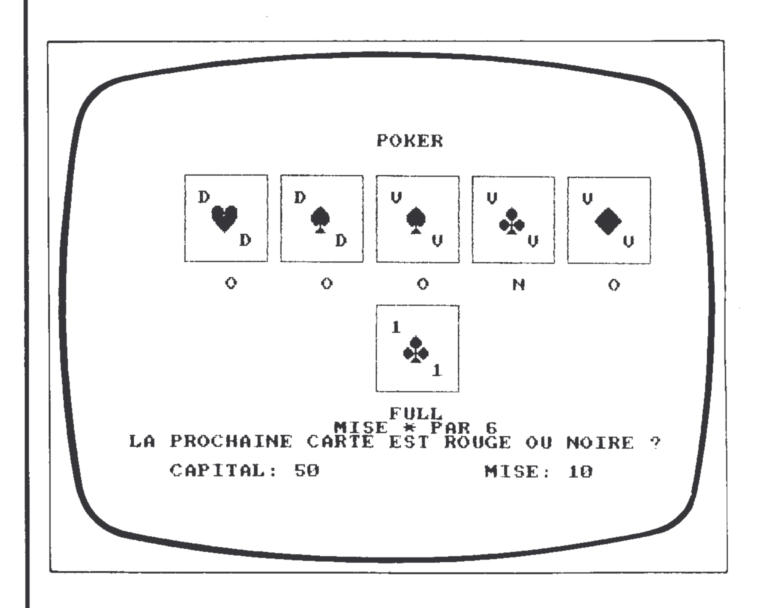
Les lignes 800 à 840 placent dans la variable C les combinaisons trouvées à partir des valeurs du tableau C (exemple : C(3)=1 et C(2)=1 donne un full donc C=5).

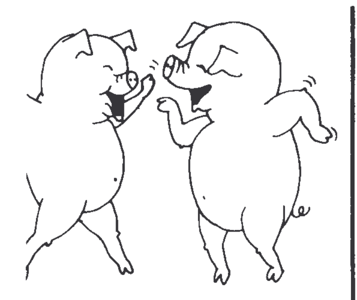
Les lignes 850 à 890 détectent la présence d'une séquence (C=4) et les lignes 900 à 920 la présence d'un flush (cinq cartes de la même couleur, C=6). Si l'on avait déjà une séquence, C=4 devient C=8 pour le flush royal.

Après totalisation des gains, on reprend une autre partie s'il reste encore du capital.

LISTE DES VARIABLES :

A$	choix du joueur
C	nombre aléatoire donnant la couleur
C()	nombre de cartes identiques
C$()	liste des combinaisons gagnantes
F	nombre aléatoire donnant la figure
G	gain du joueur
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
J$	liste des cartes
J(,)	cartes du jeu : 8 cartes de 4 couleurs
K	pointeur des caractères graphiques pour afficher le jeu
L$()	dessin des cartes
M	mise du joueur
P()	nombre de cartes identiques (ex : P(1)=3, s'il y a 3 fois la carte 7, P(2)=4 s'il y a 4 fois la carte 8)
S(,)	cartes en jeu
X et **Y**	coordonnées des cartes sur l'écran
Z	couleur de la carte à afficher



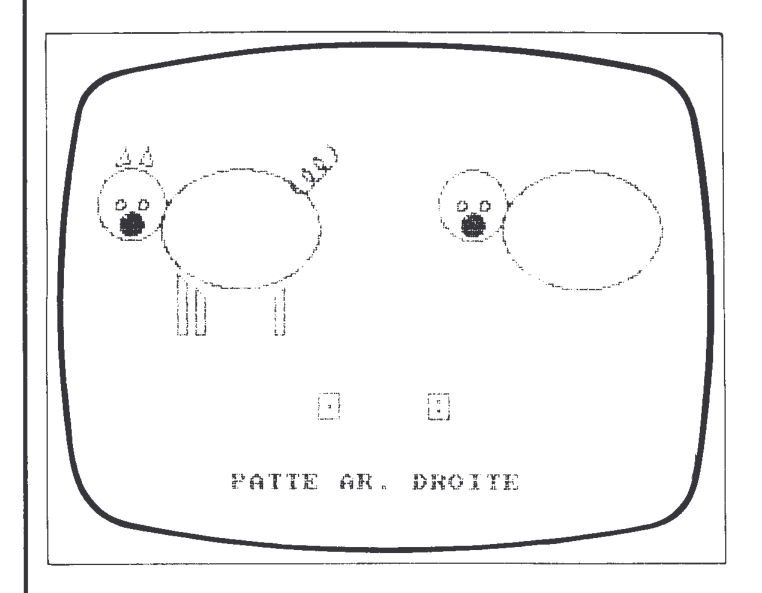


PETITS COCHONS

Plus de tricherie possible : vous pouvez jouer à deux aux célèbres petits cochons dont il faut gagner aux dés tous les éléments.

Amstrad dessine les dés sur l'écran. Chaque joueur doit les lancer en appuyant sur un des chiffres de 1 à 9 (pour forcer le sort).

Mais quelle surprise : si vous avez 9 points en tout, le corps d'un cochon apparaît. Il faudra ensuite successivement 7 points pour chaque œil, au moins un dé de six pour la queue et au moins un as pour une patte ou une oreille (un double as vous donnant droit à deux pattes).



```
10 REM PETITS COCHONS
20 REM -------------------------------
30 MODE 1:INK 0,1:INK 1,24:INK 2,16:INK 3,2:PAPER
0:PEN 1:BORDER 1
40 SYMBOL AFTER 128
50 DEG
60 RANDOMIZE TIME
70 DATA CORPS,9,OEIL DROIT,7,OEIL GAUCHE,7,QUEUE,6
80 DATA OREILLE DROITE,1,OREILLE GAUCHE,1
90 DATA PATTE AV. DROITE,1,PATTE AV. GAUCHE,1,PATT
E AR. DROITE,1,PATTE AR. GAUCHE,1
100 FOR I=1 TO 10:READ N$(I),D(I):NEXT
110 SYMBOL 128,0,0,0,24,24,0,0,0
120 SYMBOL 129,0,24,24,0,0,24,24,0
130 SYMBOL 130,96,96,0,24,24,0,6,6
140 SYMBOL 131,102,102,0,0,0,0,102,102
150 SYMBOL 132,102,102,0,24,24,0,102,102
160 SYMBOL 133,102,102,0,102,102,0,102,102
170 X(1)=200:X(2)=500
180 FOR I=1 TO 2
190 PLOT 158+I*96,162,2:DRAWR 18,0:DRAWR 0,-22:DRA
WR -18,0:DRAWR 0,22
200 NEXT
280 REM -------------------------------
290 REM LE JEU
300 FOR I=1 TO 2
310 LOCATE 1,20:PRINT CHR$(7);CHR$(18);TAB(16);"JO
UEUR";I
320 A=VAL(INKEY$)
330 IF A<1 OR A>9 THEN 320
340 FOR J=1 TO A*3
350 FOR K=1 TO 2
360 DE(K)=INT(RND*6)+1
370 LOCATE 11+K*6,16:PRINT CHR$(127+DE(K))
380 NEXT
390 NEXT
400 DES=DE(1)+DE(2)
410 J=EL(I)+1
420 IF J=4 THEN 440
430 IF D(J)=DES THEN 480
440 IF D(J)=DE(1) THEN 480
450 IF D(J)=DE(2) THEN 480
460 LOCATE 15,20:PRINT CHR$(18);"   RIEN"
470 FOR K=1 TO 500:NEXT:GOTO 590
480 EL(I)=EL(I)+1:X=X(I)
490 LOCATE 1,20:PRINT TAB(20-LEN(N$(J))/2);CHR$(18
);N$(J)
500 ON J GOSUB 800,900,1000,1700,1100,1200,1300,14
00,1500,1600
510 IF DE(1)<>1 OR DE(2)<>1 THEN 570
520 IF EL(I)=10 THEN 700
```

```
530 J=J+1
540 EL(I)=EL(I)+1
550 LOCATE 1,20:PRINT TAB(20-LEN(N$(J))/2);CHR$(18
);N$(J)
560 ON J GOSUB 800,900,1000,1700,1100,1200,1300,14
00,1500,1600
570 IF EL(I)=10 THEN 700
580 GOTO 310
590 NEXT I
600 GOTO 300
680 REM ------------------------------
690 REM FIN
700 LOCATE 1,20:PRINT CHR$(18);TAB(12);"LE JOUEUR"
;I;"GAGNE"
710 IF INKEY$<>CHR$(13) THEN 710
720 RUN
780 REM ------------------------------
790 REM CORPS
800 FOR K=1 TO 360:MOVE X-14,300:PLOTR 70*COS(K),5
0*SIN(K),2:NEXT
810 FOR K=1 TO 360 STEP 3:MOVE X-110,320:PLOTR 30*
COS(K),30*SIN(K):NEXT
820 FOR K=1 TO 360 STEP 6:MOVE X-110,304:DRAWR 10*
COS(K),10*SIN(K):NEXT
830 RETURN
880 REM ------------------------------
890 REM OEIL GAUCHE
900 FOR K=1 TO 360 STEP 6:MOVE X-120,320:PLOTR 4*C
OS(K),4*SIN(K),3:NEXT
910 RETURN
980 REM ------------------------------
990 REM OEIL DROIT
1000 FOR K=1 TO 360 STEP 6:MOVE X-100,320:PLOTR 4*
COS(K),4*SIN(K),3:NEXT
1010 RETURN
1080 REM ------------------------------
1090 REM OREILLE GAUCHE
1100 PLOT X-124,352,2:DRAW X-116,368:DRAW X-112,35
4:DRAW X-124,352
1110 RETURN
1180 REM ------------------------------
1190 REM OREILLE DROITE
1200 PLOT X-104,354,2:DRAW X-96,368:DRAW X-92,352:
DRAW X-104,354
1210 RETURN
1280 REM ------------------------------
1290 REM PATTE AVANT GAUCHE
1300 PLOT X-70,262,2:DRAW X-62,254:DRAW X-62,210:D
RAW X-70,210:DRAW X-70,262
1310 RETURN
1380 REM ------------------------------
```

```
1390 REM PATTE AVANT DROITE
1400 PLOT X-54,250,2:DRAW X-46,242:DRAW X-46,210:D
RAW X-54,210:DRAW X-54,250
1410 RETURN
1480 REM ------------------------------
1490 REM PATTE ARRIERE GAUCHE
1500 PLOT X+24,250,2:DRAW X+16,242:DRAW X+16,210:D
RAW X+24,210:DRAW X+24,250
1510 RETURN
1580 REM ------------------------------
1590 REM PATTE ARRIERE DROITE
1600 PLOT X+40,262,2:DRAW X+32,254:DRAW X+32,210:D
RAW X+40,210:DRAW X+40,262
1610 RETURN
1680 REM ------------------------------
1690 REM QUEUE
1700 FOR K=40 TO 60 STEP 10
1710 FOR L=1 TO 90 STEP 2:MOVE X+K,300+K:PLOTR 10*
COS(L),10*SIN(L),1:NEXT
1720 FOR L=180 TO 360 STEP 2:MOVE X+K,300+K:PLOTR
10*COS(L),10*SIN(L),2:NEXT
1730 NEXT
1740 RETURN
```

PROGRAMME

Lignes 30 à 200 : initialisation

Les DATA des lignes 70 à 90 définissent toutes les parties du cochon. Le chiffre qui suit le nom correspond au nombre de points à faire avec les dés.

Les lignes 110 à 160 placent dans les graphiques la représentation des six faces du dé.

Les lignes 180 à 200 dessinent les cadres autour des deux dés.

Lignes 300 à 600 : le jeu

On saisit au clavier le nombre entre 1 et 9 que tape le joueur. Ce nombre est utilisé dans la génération des dés aléatoires (boucle des lignes 340 à 390).

La variable J donne le numéro du prochain élément à afficher. On vérifie donc dans le tableau D() si les dés correspondent à cet élément. Si c'est le cas, on va à la ligne 480 pour afficher la partie du cochon. S'il s'agit d'un double as, on affiche une deuxième patte.

A chaque lancer de dés on vérifie si le cochon est complet en testant le tableau EL().

Lignes 700 à 720 : fin

Ces lignes affichent le nom du gagnant et recommencent une partie dès qu'une touche est appuyée.

Lignes 800 à 1750 : affiche les cochons

Chaque sous-programme affiche une partie du cochon en haute-résolution à droite ou à gauche de l'écran suivant la valeur de la variable X.

LISTE DES VARIABLES :

A	valeur de la touche appuyée
D()	valeur de chaque élément du cochon
DE()	valeur des deux dés
DES	somme des deux dés
EL()	nombre d'éléments déjà affichés
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
K	indice de boucle FOR/NEXT
N$()	noms des éléments du cochon
X()	position du cochon sur l'écran

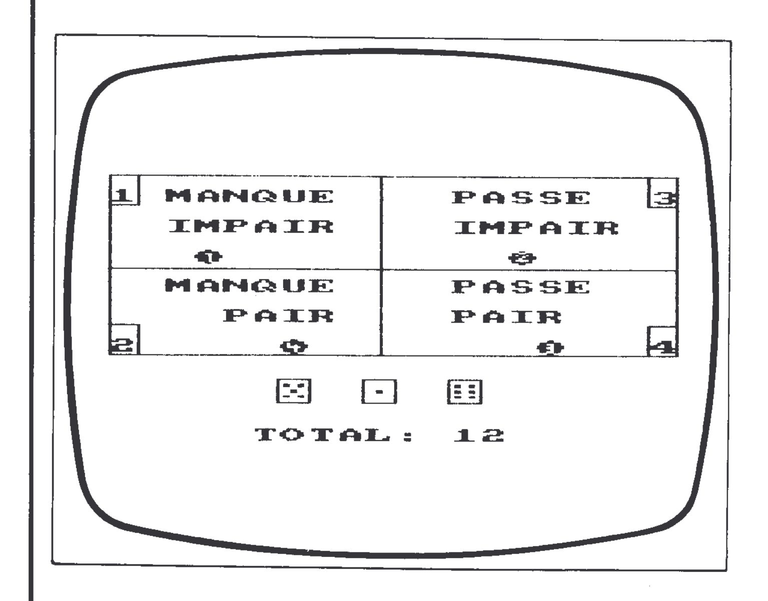


PASSE-DIX

Le passe-dix se joue avec trois dés et un tapis de quatre cases. Chaque joueur (jusqu'à quatre) mise à chaque partie sur une des cases en donnant son numéro. Si la somme des dés correspond à un des nombres de la case choisie, le joueur remporte le double de sa mise.

Les nombres dits « MANQUE » sont ceux compris entre 3 et 10, et les « PASSE » ceux entre 11 et 18 (d'où le nom du jeu). Les MANQUE et les PASSE sont séparés pairs et impairs pour obtenir quatre cases.

Comme la partie se joue sur dix tirages, chaque joueur a un capital de 10 jetons dès le début. Le programme affiche les résultats en fin de partie.



```
10 REM PASSE-DIX
20 REM ------------------------------
30 DEFINT A-Z
40 CLS
50 INPUT"NOMBRE DE JOUEURS ";N
60 IF N<1 OR N>4 THEN PRINT"ENTRE 1 ET 4 !":GOTO 5
0
70 MODE 0:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1
80 SYMBOL AFTER 128
90 RANDOMIZE TIME
100 SYMBOL 128,0,0,0,24,24,0,0,0
110 SYMBOL 129,0,24,24,0,0,24,24,0
120 SYMBOL 130,96,96,0,24,24,0,6,6
130 SYMBOL 131,102,102,0,0,0,0,102,102
140 SYMBOL 132,102,102,0,24,24,0,102,102
150 SYMBOL 133,102,102,0,102,102,0,102,102
160 SYMBOL 134,24,126,102,247,247,118,118,24
170 SYMBOL 135,24,102,90,247,239,66,126,24
180 SYMBOL 136,24,126,102,247,231,118,102,24
190 SYMBOL 137,24,126,94,215,195,118,118,24
200 C(1)=8:C(2)=13:C(3)=3:C(4)=15
210 X(1)=3:X(2)=3:X(3)=13:X(4)=13
220 Y(1)=6:Y(2)=12:Y(3)=6:Y(4)=12
230 FOR I=1 TO N:P(I)=10:NEXT
380 REM ------------------------------
390 REM DESSIN DU JEU
400 FOR K=1 TO 10
410 CLS
420 LOCATE 1,2:PRINT"1":LOCATE 3,2:PRINT"MANQUE":L
OCATE 3,4:PRINT"IMPAIR"
430 LOCATE 20,2:PRINT"3":LOCATE 13,2:PRINT"PASSE":
LOCATE 13,4:PRINT"IMPAIR"
440 LOCATE 1,12:PRINT"2":LOCATE 3,8:PRINT"MANQUE":
LOCATE 5,10:PRINT"PAIR"
450 LOCATE 20,12:PRINT"4":LOCATE 13,8:PRINT"PASSE"
:LOCATE 13,10:PRINT"PAIR"
460 PLOT 2,398:DRAWR 634,0:DRAWR 0,-192:DRAWR -634
,0:DRAWR 0,192
470 PLOT 304,398:DRAWR 0,-192
480 PLOT 2,298:DRAWR 634,0
490 PLOT 2,366:DRAWR 32,0:DRAWR 0,32:PLOT 636,366:
DRAWR -32,0:DRAWR 0,32
500 PLOT 2,238:DRAWR 32,0:DRAWR 0,-32:PLOT 636,238
:DRAWR -32,0:DRAWR 0,-32
510 FOR I=0 TO 192 STEP 96
520 PLOT 190+I,180:DRAWR 36,0:DRAWR 0,-26:DRAWR -3
6,0:DRAWR 0,26
530 NEXT
540 FOR I=1 TO N
550 LOCATE 4,21+I:PRINT"JOUEUR";I;":";P(I)
```

```
560 NEXT
680 REM ------------------------------
690 REM LE JEU
700 FOR I=1 TO N
710 LOCATE 7,18:PRINT "JOUEUR";I:PRINT TAB(3);"NUM
ERO DE LA CASE ?"
720 J(I)=VAL(INKEY$):IF J(I)=0 THEN 720
730 IF J(I)<1 OR J(I)>4 THEN 720
740 LOCATE X(J(I))+I,Y(J(I)):PRINT CHR$(I+133)
750 LOCATE 1,18:PRINT CHR$(18):PRINT:PRINT CHR$(18
)
760 NEXT
770 FOR I=1 TO RND*20+20
780 FOR J=1 TO 3
790 D(J)=INT(RND*6)+1
800 LOCATE 4+J*3,15:PRINT CHR$(127+D(J))
810 NEXT
820 NEXT
830 D=D(1)+D(2)+D(3)
840 LOCATE 6,18:PRINT"TOTAL:";D
850 IF INKEY$<>CHR$(13) THEN 850
860 IF D>10 THEN J=3 ELSE J=1
870 IF D/2=INT(D/2) THEN J=J+1
880 FOR I=1 TO N
890 IF J(I)=J THEN P(I)=P(I)+1 ELSE P(I)=P(I)-1
900 NEXT
910 NEXT
1080 REM ----------------------------
1090 REM RESULTATS
1100 MODE 1
1110 PRINT TAB(15);"RESULTATS"
1120 FOR I=1 TO N
1130 LOCATE 9,10+I*2:PRINT"JOUEUR";I;":";P(I);"JET
ON(S)"
1140 NEXT
1150 IF INKEY$<>CHR$(13) THEN 1150
1160 RUN
```

Programme

Lignes 30 à 230 : initialisation

Après la saisie du nombre de joueurs, le programme initialise les nombres aléatoires (ligne 90).

Les lignes 100 à 190 initialisent les graphiques avec le dessin des six faces d'un dé et des quatre jetons.

Lignes 400 à 560 : dessin du jeu

On commence par afficher le nom des cases (lignes 420 à 450), puis on trace le tapis (lignes 460 à 500).

Lignes 700 à 910 : le jeu

C'est la boucle des lignes 700 à 760 qui saisit le choix des joueurs.

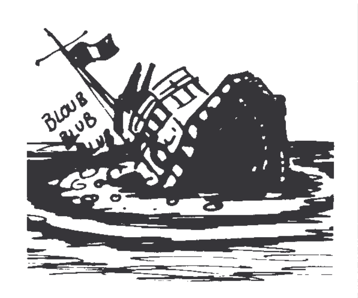
Les dés sont affichés par la boucle des lignes 770 à 820 en fonction des nombres aléatoires contenus dans le tableau D.

On calcule en fonction de la somme des trois dés (variable D) le numéro de la case gagnante (variable J).

Il ne reste plus qu'à comparer la variable J avec le choix de chaque joueur (tableau J). On incrémente le nombre de points des joueurs gagnants (tableau P) et l'on décrémente celui des perdants.

Liste des variables :

D	somme des trois dés
D()	valeur des trois dés
I	indice de boucle FOR/NEXT
J	numéro de la case gagnante
J()	choix des joueurs
K	indice de boucle FOR/NEXT
N	nombre de joueurs
P()	nombre de jetons restant à chaque joueur
X() et **Y()**	coordonnées des cases sur l'écran



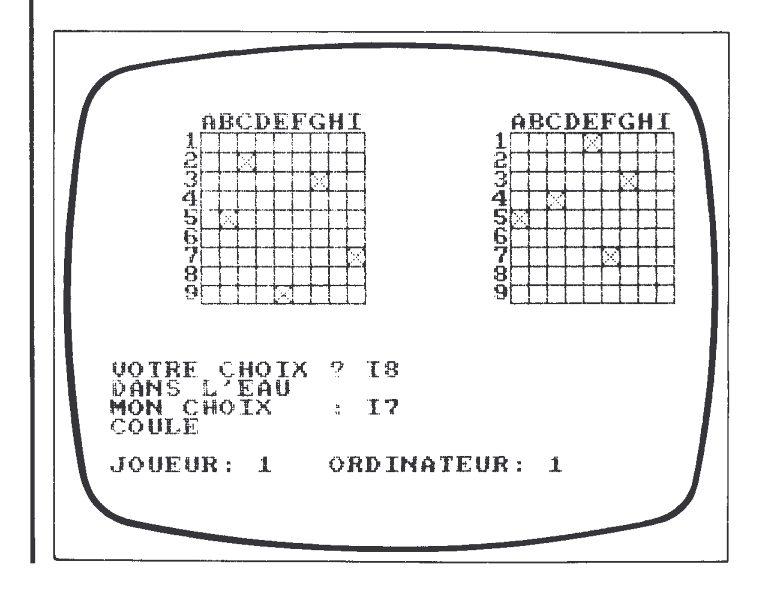
BATAILLE NAVALE

Serez-vous assez rapide pour détruire la flotte de votre ennemi avant qu'il ne découvre les emplacements de vos sous-marins ?

Au lancement du programme, vous devez situer la position de vos cinq submersibles sur une grille de neuf colonnes (de A à I) sur neuf lignes (de 1 à 9) sous la forme A1 ou C3. Cette grille s'affiche alors à gauche de l'écran.

La bataille commence. Vous réfléchissez et donnez les coordonnées supposées d'un sous-marin ennemi. Si vous avez touché la cible, l'ordinateur signale COULE, sinon DANS L'EAU. Dans les deux cas il coche la case correspondante sur la grille qui apparaît à droite de l'écran. L'ordinateur propose à son tour une position. S'il saborde un de vos bâtiments, Amstrad l'annonce et l'efface de la grille de gauche.

Après votre coup et celui de l'ordinateur, le score des deux adversaires apparaît. Amstrad vous demandera un autre coup lorsque vous aurez appuyé sur ENTER et ceci jusqu'à ce que l'une des flottes soit totalement anéantie.



```
10 REM BATAILLE NAVALE
20 REM ------------------------------
30 DIM J(9,9,4)
40 MODE 1:INK 0,1:INK 1,24:INK 2,6:INK 3,15:PAPER
0:PEN 1:BORDER 1
50 RANDOMIZE TIME
60 PRINT"ENTREZ LES COORDONNEES DES 5 SOUS-MARINS"
70 FOR I=1 TO 5
80 PRINT"SOUS-MARIN";I;
90 INPUT A$:A$=A$+"  "
100 X=ASC(A$)-64:Y=VAL(MID$(A$,2,1))
110 IF X<1 OR X>9 OR Y<1 OR Y>9 THEN 80
120 IF J(X,Y,1)<>0 THEN 80
130 J(X,Y,1)=1
140 NEXT
150 FOR I=1 TO 5
160 X=INT(RND*9)+1:Y=INT(RND*9)+1
170 IF J(X,Y,3)<>0 THEN 160
180 J(X,Y,3)=1
190 NEXT
280 REM ------------------------------
290 REM DESSIN DU JEU
300 CLS
310 X=5:T=1:GOSUB 800
320 X=22:T=2:GOSUB 800
330 LOCATE 1,15:PRINT CHR$(18):PRINT CHR$(18):PRIN
T CHR$(18)
380 REM ------------------------------
390 REM TOUR DU JOUEUR
400 LOCATE 1,14:PRINT CHR$(18);"VOTRE CHOIX ";
410 INPUT A$:A$=A$+"  ":X=ASC(A$)-64:Y=VAL(MID$(A$
,2,1))
420 IF X<1 OR X>9 OR Y<1 OR Y>9 THEN PRINT CHR$(7)
;:GOTO 400
430 IF J(X,Y,3)=1 THEN PRINT"COULE":J(X,Y,3)=0:T1=
T1+1 ELSE PRINT"DANS L'EAU"
440 J(X,Y,2)=1
450 PLOT (22+X-1)*16,398-Y*16,3:DRAW (22+X)*16,382
-Y*16:PLOT (22+X)*16,398-Y*16:DRAW (22+X-1)*16,382
-Y*16
480 REM ------------------------------
490 REM TOUR DE L'ORDINATEUR
500 PRINT"MON CHOIX   : ";
510 X=INT(RND*9)+1:Y=INT(RND*9)+1
520 IF J(X,Y,4)=1 THEN 510
530 PRINT CHR$(X+64);RIGHT$(STR$(Y),1)
540 IF J(X,Y,1)=1 THEN PRINT"COULE":J(X,Y,1)=0:T2=
T2+1 ELSE PRINT"DANS L'EAU"
550 J(X,Y,4)=1
560 PRINT:PRINT"JOUEUR:";T1;"  ORDINATEUR:";T2
```

```
570 PLOT (5+X-1)*16,398-Y*16,0:DRAW (5+X)*16,382-Y
*16:PLOT (5+X)*16,398-Y*16:DRAW (5+X-1)*16,382-Y*1
6
580 IF INKEY$="" THEN 570
680 REM -------------------------------
690 REM FIN DE PARTIE
700 IF T1<5 AND T2<5 THEN 330
710 PRINT
720 IF T1=5 AND T2=5 THEN PRINT"EXAEQUO":GOTO 740
730 IF T1=5 THEN PRINT"VOUS GAGNEZ" ELSE PRINT"JE
GAGNE"
740 IF INKEY$<>CHR$(13) THEN 740
750 RUN
780 REM -------------------------------
790 REM DESSIN D'UN CADRE
800 PEN 1
810 FOR J=16 TO 148 STEP 16
820 PLOT X*16,382,1:DRAW J+X*16,382:DRAW J+X*16,38
2-J:DRAW X*16,382-J:DRAW X*16,382
830 PLOT 144+X*16,238:DRAW 144+X*16-J,238:DRAW 144
+X*16-J,238+J:DRAW 144+X*16,238+J:DRAW 144+X*16,23
8
840 NEXT
850 PEN 2
860 FOR I=1 TO 9
870 LOCATE X+I,1:PRINT CHR$(I+64)
880 LOCATE X,1+I:PRINT CHR$(I+48)
890 NEXT I
900 PEN 1
910 FOR I=1 TO 9
920 FOR J=1 TO 9
930 IF J(I,J,T)=0 THEN 950
940 PLOT (X+I-1)*16,398-J*16,3:DRAW (X+I)*16,382-J
*16:PLOT (X+I)*16,398-J*16:DRAW (X+I-1)*16,382-J*1
6
950 NEXT:NEXT
960 RETURN
```

Programme

Lignes 30 à 190 : initialisation

Les lignes 60 à 140 saisissent les coordonnées des sous-marins du joueur. Ce sont les lignes 150 à 190 qui placent celles de l'ordinateur grâce aux nombres aléatoires X et Y.

Lignes 300 à 330 : dessin du jeu

Pour le dessin du jeu, on fait appel au sous-programme en 800 qui affiche une grille aux coordonnées X et Y.

Lignes 400 à 450 : tour du joueur

On saisit dans X et Y les coordonnées du tir. La ligne 430 vérifie si un sous-marin est coulé.

Lignes 500 à 580 : tour de l'ordinateur

Identique au tour du joueur, mais les coordonnées sont déterminées par deux nombres aléatoires (variables X et Y).

Lignes 700 à 750 : fin de partie

On affiche un des trois messages possibles en fonction des résultats de la partie.

Liste des variables :

A$	réponse du joueur
I	indice de boucle FOR/NEXT
J	indice de boucle FOR/NEXT
J(,,)	coordonnées des bateaux et des coups joués
T	numéro de la grille à afficher
T1	points du joueur
T2	points de l'ordinateur
X et **Y**	coordonnées du tir

EN CAS D'ERREUR

Tous les listings de cet ouvrage sont des originaux sortis directement de l'imprimante. Tous les programmes de jeu proposés ici ont été testés attentivement sur Amstrad CPC 464 et 664. Une erreur serait donc exceptionnelle.

Si, malgré tout, l'un des programmes ne fonctionnait pas, assurez-vous d'avoir correctement recopié le listing en Basic.

Voici quelques conseils qui vous aideront à déceler l'erreur :
- ne confondez pas Ø (zéro) et O (lettre) ;
- assurez-vous d'avoir placé correctement les "." et ";" nécessaires ;
- comptez les lignes du programme à recopier et faites attention à ne pas en oublier ;
- donnez toujours le même nom à vos variables d'un bout à l'autre du programme.

Et surtout... Armez-vous de courage !

SUPER JEUX AMSTRAD

50 programmes de jeux en Basic

50 super programmes de jeux d'adresse, de réflexion et de hasard vous proposent de maîtriser rapidement le BASIC de votre Amstrad CPC 464 ou 664.

En plus du plaisir que vous aurez à déjouer les pièges tendus par les "Motos lumineuses", à trouver le meilleur chemin dans le "Labyrinthe 3D", ou à gagner au "Tiercé", vous apprendrez, en jouant, à construire des programmes de plus en plus complexes en vous aidant des commentaires pédagogiques de l'auteur, et de sa précieuse liste de variables.

"Super Jeux Amstrad" contient des programmes faciles à rentrer et à modifier selon votre imagination, ainsi que de très belles pages-écran.

Possesseurs d'Amstrad, à vos claviers!





Editions du P.S.I.
BP 86
FRANCE - 77402 LAGNY/MARNE CEDEX
ISBN 2.86595.257.6

120 FF

Couverture GIBRAT